OPTIMIZATION OF THE POLYCONDENSATION STEP OF POLYETHYLENE TEREPHTHALATE FORMATION IN SEMIBATCH REACTORS

A Thesis Submitted

In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

by
A. E. SAINATH

to the

DEPARTMENT OF CHEMICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
OCTOBER, 1984
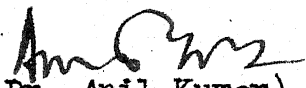
CERTIFICATE

This is to certify that the present work on "OPTIMIZATION OF THE POLYCONDENSATION STEP OF POLYETHYLENE TEREPHTHALATE FORMATION IN SEMIBATCH REACTORS" has been carried out under my supervision and that it has not been submitted elsewhere for a degree.

October 10,1984

(Dr. Anil Kumar)
Professor
Department of Chemical Engineering
Indian Institute of Technology
Kanpur-208016, U.P.
India

ACKNOWLEDGEMENTS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	Acetaldehyde
a	moles of acetaldehyde
A_i	Frequency factor, lit/mole-min
D*	Desired amount of diethylene glycol in the polymer
E_g, E_c, D, E_v	Functional groups of polymer molecules
e_g, e_c, d, e_v	Moles of E_g, E_c, D, E_v
E_i	Activation energy, Kcal/mole
E_G	Ethylene glycol
f_i	Functions of e_g, e_c, d, e_v, g,w,a and temperature
$\underline{f}$	vector having f_i as components
g	Moles of ethylene glycol
H	Hamiltonian function
I	Objective function
J	Term inside the integral in the objective function
K_3, K_4, K_5	Equilibrium constants
k_i	Rate constants, lit/mole.min
M_P	Moles of polymer
n	Chain length of linear oligomer
$\underline{P}$	Symmetric matrix of the order (4x4) in the Riccati equation
$P_{11}, P_{12}, \ldots \ldots P_{44}$	Elements of $\underline{P}$
$\overset{\circ}{\underline{P}}$	Time derivative of $\underline{P}$

P_T — Total pressure, mm Hg

P_G^o, P_W^o, P_A^o — Vapour pressures of ethylene glycol, water and acetaldehyde, atm.

Q_{EG}, Q_W, Q_A — Rate of flashing of ethylene glycol, water, and acetaldehyde from the reaction mass, moles/hr

$\underline{q}$ — (4x1) vector function of time

$\underline{q}^o$ — Time derivative of $\underline{q}$

R — Universal gas constant, cal/mole °K

t — Time, hr

t_f — Final polymerization time, hr

T — Temperature, °K

V — Volume of the reaction mass, litres

V_P, V_G, V_A — Molar volumes of polymer, ethylene glycol, and acetaldehyde, litres/mole

W — Water

$\underline{X}$ — State variable vector

X_i — State variables

X_i^o — Time-derivative of state variables X_i

X_G, X_W, X_A — Mole fractions of ethylene glycol, water, and acetaldehyde in the reaction mass

Z — Total amount of reacted bonds, moles

Greek letters

μ_n — Number-average molecular weight of the polymer

μ_{nd} — Desired number-average molecular weight of the polymer

η — Viscosity of the reaction mass, poise

α_1 — Weighting parameter for diethylene glycol content, (e_D) in the polymer

α_2 — Weighting parameter for the number-average molecular weight of the polymer

α_3 — Weighting parameter for side products formation

$\underline{\lambda}$ — Adjoint variable vector

λ_i — Adjoint variables

ε — Step-size factor deciding the magnitude of correction in temperature in equations (17) and (34)

Δt — Width of the time interval, hr

$\delta \underline{X}$ — Perturbation vector of state variables X_i

$\delta \underline{X}'$ — Time derivative of $\delta \underline{X}$

δT — Perturbation in temperature

$\delta \underline{\lambda}$ — Perturbation vector of adjoint variables λ_i

Superscripts

T — Transpose

j — Iteration number

Subscript

o — Feed conditions

ABSTRACT

The polycondensation stage of polyethylene terephthalate (PET) formation is assumed to include side reactions leading to the formation of diethylene glycol, vinyl end groups, acid end groups, in addition to the usual polymerization of bis (2- hydroxyethyl) terephthalate (BHET) in semi-batch reactors. A flexible objective function has been proposed with temperature as well as pressure as control variables. Computations show that the pressure should be reduced to the lowest limit under all possible conditions. Consequently optimal temperature profiles in batch reactors are obtained for various lower limits of reactor pressures using the combination of first and second variation techniques. For the first variation technique, the vector iteration method of computation was used and the near optimal profile so obtained was used as the initial guess for the second variation technique.

The result of optimization shows that the lower limit of pressure and weighting parameters appearing in the objective function have profound effect on the optimum profiles. For higher pressures, one obtains that a high temperature must be used initially which must be lowered later on to minimize the formation of side products. However for lower pressures, the temperature must be increased from a low value initially; however for large times of polymerization, this must be lowered to minimize the formation of side products. It is thus seen that the optimum temperature profile exhibits a broad maximum.

CHAPTER 1

INTRODUCTION

Among the saturated polyesters, polyethylene terephthalate

(PET, $\left[CH_2CH_2 - O - \overset{O}{\underset{}{C}} - \langle o \rangle - \overset{O}{\underset{}{C}} - O \right]_n$) is the most important

polymer from commercial point of view. It is also known by

various other names such as "Dacron" and "Terylene". Fibre-

grade PET is mainly used for making textiles[1,2] and its

contribution to the world synthetic fibre demand is around

forty percent. It has many other industrial applications, for

example it can be used as a raw material for the production

of a new transparent film for electrical insulation and also as

a base for a new photographic film. It can also be used as a

moulding material.

Method of PET Production:

In Industries, the production of fibre-grade PET is

carried out in four stages.[3] These are: (1) Transesterification

or direct esterification, (2) Prepolymerization, (3) Melt

polycondensation, and (4) Solid state polycondensation. The

different stages are as shown in Fig. 1. The monomer for the

production of PET is bis (hydroxyethyl) Terephthalate (BHET,

$HOCH_2CH_2O\overset{O}{\underset{}{C}} - \langle o \rangle - \overset{O}{\underset{}{C}}OCH_2CH_2OH$). In the first stage, this

monomer is synthesized either through a transesterification

or a direct esterification reaction, using Dimethyl Terephthalate

(DMT, $CH_3O\overset{O}{\underset{}{C}} - \langle o \rangle - \overset{O}{\underset{}{C}}OCH_3$) and Ethylene Glycol (EG, $OHCH_2CH_2OH$)

as the raw materials. In transesterification, DMT and EG are reacted and the condensation product, methanol is continuously removed, whereas in direct esterification, terephthalic acid (TPA) is employed in place of DMT, and the reaction takes place between this and EG[4,5], with the evolution of water as the condensation product, which is removed from the reaction mass. Nowadays, Industries are switching to direct esterification instead of transesterification because the former route appears to offer some advantages[6] over the other route and also since pure fibre-grade TPA is available. The main advantage is that the step of manufacturing Ethylene Glycol from ethylene oxide can be eliminated since TPA can be directly reacted with ethylene oxide[7], instead of ethylene glycol. It should also be noted that there are some disadvantages in using Ethylene oxide as the raw material since it is explosive and may cause serious safety hazards. In the second stage, which is sometimes called the polycondensation stage, BHET (produced from the first stage) is polymerized in a prepolymerization reactor upto a μ_n of approximately 30 (or upto about 95% conversion). In this stage, the overall reaction is operated in reaction controlled regime[8-11] and the viscosity of the reaction mass increases approximately to 50 poise.[12] In the third stage or in the final stages of polycondensation, the polymer from the second stage is further polymerized to a
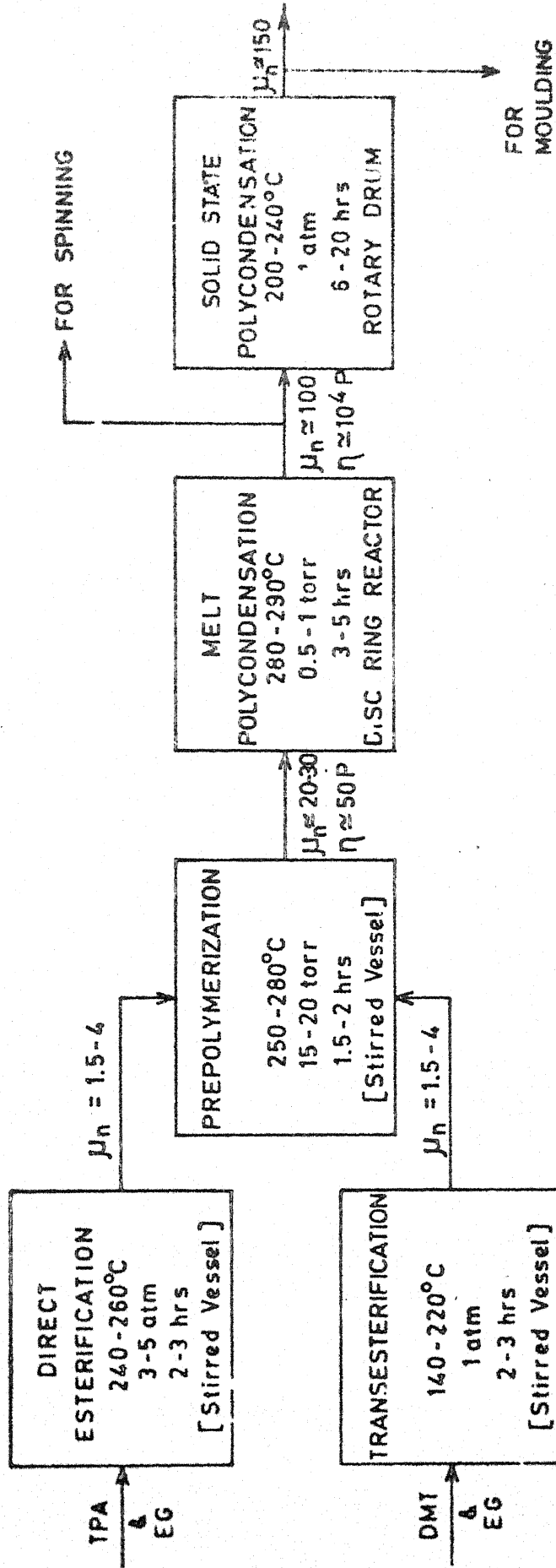
Fig. 1 Different stages involved in the production of PET.

μ_n of approximately 100 using vacuum and a special agitation
system. At the end of this stage, the viscosity of the polymer
melt is of the order of few thousand poise. Due to this high
viscosity, the diffusion of condensation product becomes
difficult and hence mass-transfer effects should also be
accounted in the overall rate of polymerization reaction.
Under these conditions, it is necessary to carry out the
reaction in special reactors and usually wiped-film reactors[13-17]
which have facility of mechanical generation of interfacial
area are used in the continuous process. In the batch process,
a single reactor with conventional helical screw or ribbon
agitators[18] can be used in the final stages of polycondensation.
In the fourth stage or the solid state polycondensation stage,[19]
the resulting polymer from the final stages of polycondensation
is solidified and prepared in the form of small chips.
Using these chips, the process is carried out at a temperature
which is in between the glass transition temperature and the
melting temperature of the polymer.

Chemistry of PET Formation:

In the second stage or the polycondensation stage of
PET formation, the main reaction taking place is the poly-
condensation reaction given by reaction (1) in Table 1.[2, 5]
Several side reactions are known to take place in addition to
the above main reaction.[5,7,10,11] These are: (1) Redistribution

reaction, (2) Cyclization reaction,(3) Diethylene glycol formation, (4) Acid end group formation and (5) Vinyl and group formation. The last three reactions are also given in Table 1. Due to these side reactions, water (w) and Acetal-dehyde (A) are also formed in addition to EG. Due to a number of reactions taking place simultaneously, the reaction mass consists of various functional groups which are also summarized in Table 1. The growth of the polymer chains takes place due to the main reaction in Table 1 and the quality of the polymer formed is strongly dependent on the amount of various side products formed. It is therefore, important to control these side products.[20]

Mechanism of Catalysis:

In the synthesis of PET, the rate constants for the polycondensation and degradation reactions are dependent upon the catalyst used in the process. Since the various properties of PET are dependent upon the rates of these reactions, the selection of a suitable catalyst is important.

In the transesterification process, the catalysts commonly employed are acetates of zinc, manganese, calcium, sodium, lead and magnesium. The relative efficiency of these catalysts has been studied[21] and it is found that zinc acetate is the most effective. But in Industries, usually a mixture of these catalysts is used and the catalyst with lower efficiency

(sodium acetate) serves as a 'tracer' in many cases. In the direct esterification process, the proton formed during the dissociation of carboxyl group serves as a catalyst. For the polyeondensation reaction, the catalyst generally used is Antimony trioxide or Antimony acetate. Two types of catalysts are used in the transesterification and polycondensation reactions because these catalysts have their own medium dependence. For example metal acetates are very active in a high and low hydroxyl content medium[22] but are easily poisoned by even small amounts of acid and groups[22,23] in the reaction mass, whereas the activity of antimony trioxide or antimony acetate is not affected by acid groups, but it is known to increase as the concentration of hydroxyl end groups decreases.[22,24] The mechanism of catalysis is very important for a better understanding of the catalyst performance.

Transesterification Catalysis:

The rate of transesterification reaction depends on the concentration of a soluble metal alcoholate which is formed according to the following reaction:

$$M(CH_3CO)_2 + 2ROH \rightleftharpoons M(OR)_2 + 2CH_3CO_2H \qquad (1)$$
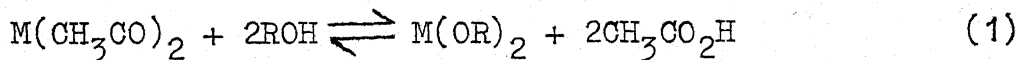
Under certain operating conditions in the reactor, the formation of this active alcoholate can be favoured by removing the acetic acid formed in the reaction. The rate of transesterification reaction is reduced if the DMT used in the

reaction is not pure i.e. if it contains any nonvolatile carboxylic acids such as TPA as an impurity or if there are any acid end groups present in the reaction mass formed as a result of side reaction. The mechanism proposed[23] for trans-esterification reaction employing metal ion catalysis is shown in Fig. 2(a).

Polycondensation Catalysis:

In polycondensation catalysis the first step consists of the coordination of the metal ion to the ester carbonyl bond which enhances the polarity of this bond and thereby facilitates the nucleophilic attack.[25] The polarizing action of a metal ion is a function of its charge as well as ionic radius and the catalytic activity increases if the cationic charge increases and the radius decreases. The metal ion can even build the nucleophilic agent in a complex[26,27] so that its attack on the carbonyl carbon atom that has become positive is made easier. The polycondensation reaction proceeds by the activated hydroxyl group on the carbon of the carbonyl group coordinated to the metal as shown in Fig.2(b).

Due to the coordination of antimony compounds with hydroxyl groups, they cannot react with carbonyl groups of esters[24] and hence their activity is less in the early stages of polycondensation. When the reaction proceeds, the concentration of the hydroxyl groups decreases and hence the catalyst

R^0, R^1, R^2, R^3 — ALKYL GROUPS
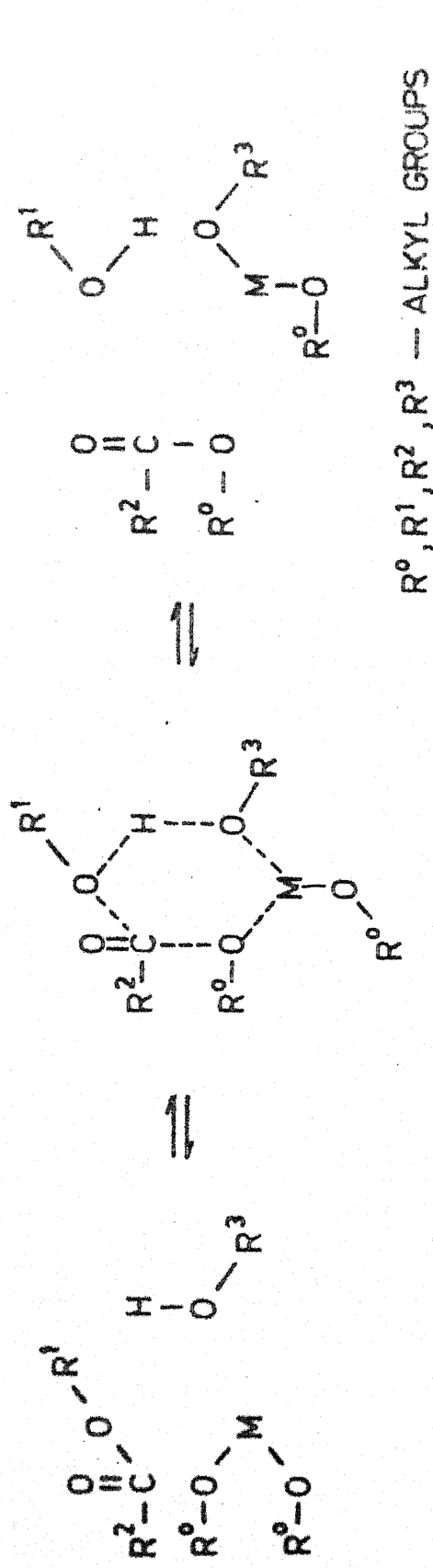
Fig. 2(a) Mechanism proposed for transesterification reaction employing metal ion catalysis.
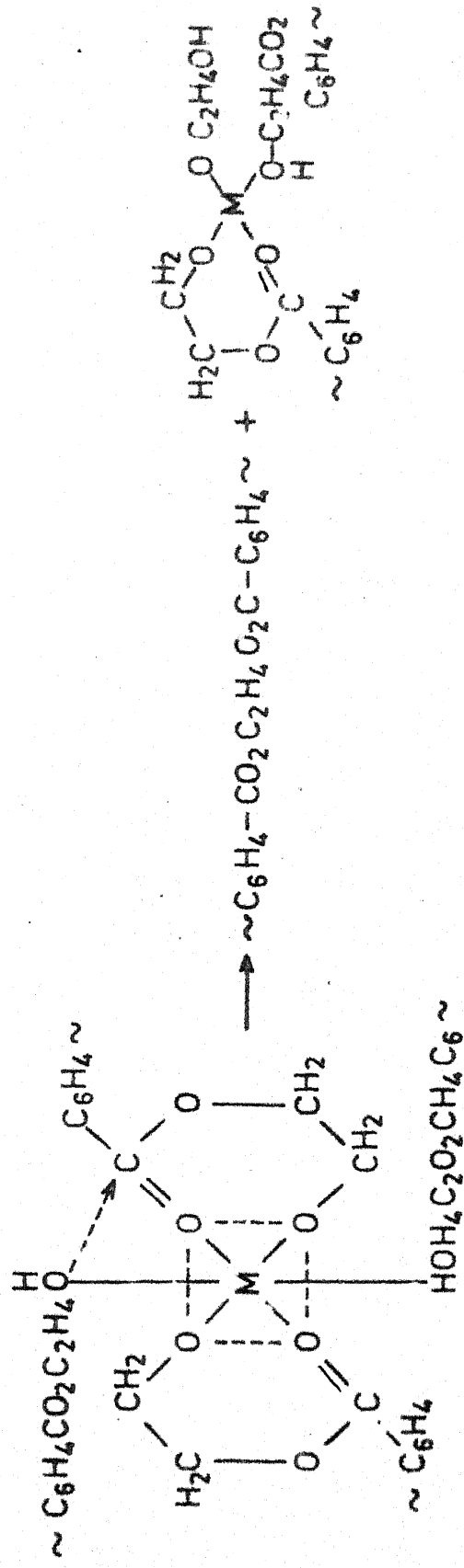
Fig. 2(b) Mechanism proposed for polycondensation reaction employing metal ion catalysis.

activity is improved. By decreasing the melt thickness,
the efficiency of the removal of volatile component (EG) can
be improved, thereby improving the availability of the catalyst
to the polycondensation reaction which increases the rate of
polycondensation reaction.[28]   Tomita[29,30] found that the
stability constant of dibenzoyl methane complex of each metal
species is critical in determining the catalytic activity of
metal compounds in transesterification, polycondensation and
thermal degradation of PET. Metal compounds having stability
constants around 12 were found to be more effective as catalysts
in the polycondensation stage.

From the reaction mechanism proposed, it is obvious that
during the polymerization there is a molecular weight distri-
bution of oligomers due to the several side reactions taking
place in the polycondensation stage of PET formation in addition
to the main reaction . Even in the case of first stage reactors
it has been shown that several side reactions take place along
with the BHET formation. But the amount of these side products
formed is a very small quantity and it can be safely assumed
that the feed to the polycondensation stage is BHET alone even-
though some recent studies[31,32] believed that these cannot be
neglected for any true simulation.   In order to get the detailed
molecular weight distribution of oligomers, all the side reactions
must be taken into consideration and if this is done, the result

is a set of highly nonlinear differential equations which cannot be easily solved even on a computer. The only way to avoid this computational difficulty is by carrying out the analysis of the reactor in terms of functional groups, even though it has been shown in the case of PET that one cannot do this due to cyclization and redistribution reactions. According to Kumar et al.[9-11], if these two side reactions are included, the reactor analysis should be carried out in terms of individual polymer molecules instead of functional groups. But the simulation studies of Ravindranath[5,8] and Mashelkar confirmed that the cyclization reaction can be neglected. Also the redistribution reaction can be neglected since it affects the second and higher moments of functional groups. Hence, in our kinetic model, we neglected the cylization and redistribution reactions.

In the first stage of PET production, the usual molar ratio of DMT and EG in the feed is about 1:2 and the reaction occurs mostly in the temperature range of 140-200°C at one atmosphere pressure. The use of a catalyst is common so that the reaction is completed in a reasonable residence time and it is already mentioned that zinc acetate is the most effective catalyst. PET is formed when the product from the first stage reactor is heated to 270 - 285°C with continuous evacuation to low pressures[3,4] (2-1 mmHg). The high temperature is required in order to maintain the polyester product in molten

form. The growth of the polymer molecules is accompanied
by a substantial increase in the fluid viscosity and extreme
diffusional resistances are encountered by ethylene glycol (EG)
and methanol species if the polymer is solidified partially.
Low pressures are therefore necessary to drive the polymeri-
zation reaction to completion. It is necessary to vary the
temperature and pressure within the reactor as polymerization
proceeds and the temperature and pressure histories normally
used in Industry are given by the following equation:

$$t = 200 - 50 \, e^{-1.2\Theta} \qquad \qquad (a)$$

$$P_T = 760 \exp[-1.859 \, \Theta + 1.776 \, \Theta^2 - 1.253 \, \Theta^2] \qquad (b)$$

$$(2)$$

where t is the temperature in °C, P_T is the total pressure
in mmHg, and Θ is the residence time of the reactor in hours.
From the above equations it is clear that the temperature
should be continuously increased whereas the pressure decreased
as the polymerization reaction proceeds.

Even though PET is an important fibre from commercial point
of view, only few optimization studies are available in the
literature. Based on the kinetic studies of Challa[33] and
others, Ault and Mellichamp[4] optimized the two-stage production
of PET. Their objective in the first stage was to convert DMT
through ester exchange with EG to BHET. In the second stage

the pressure was reduced to drive off the excess EG. They
used metal acetate and Antimony trioxide as catalysts in the
first and second stages. The reaction temperatures used were of
the order of 150-200°C. They found that the total reacting
volume decreased by 25% during the course of the first stage
and their studies suggest that the reaction time should be in
between 25 min and 125 min as equilibrium conditions were
attained at the latter time. They also recommended a reaction
time of 70 min. for the second stage. Further studies on the
optimization of PET reactors include those of Kumar et al.[31,32]
These workers proposed an objective function which would be
relevant to designing a new plant.[34] In the first stage reactor,
their objective was to maximize the conversion of DMT while
simultaneously minimizing the amount of various side products
formed. In the second stage reactor, their objective was to
bring the μ_n of the polymer to a desired value while simulta-
neously minimizing the amount of various side products formed.
The proposed objective function included three weighting
parameters which takes values depending on what is required.
These weighting parameters were shown to have considerable
influence on the optimal temperature profiles with some values
of these weighting parameters, the optimal temperature profiles
obtained were of the same form as calculated using Eq.(2). In
this work, we used the same objective function.

In polycondensation reactions, pressure is an important variable besides temperature because the flashing of the volatile components, ethylene glycol, water and acetaldehyde, depends upon the pressure applied in the reactor. In Industrial reactors, both pressure and temperature are independent variables. Earlier studies[31,32] considered temperature as the control variable only and computed optimal temperature profiles using various reactor pressures. We first considered both temperature and pressure as the control variables and carried out optimization studies for this two-dimensional problem using the gradient method. Preliminary computations showed that the pressure falls first and settles down on its preset lower limit and only after this has happened, the temperature profile begins to be adjusted. These results suggest that the polymerization should be carried out at the lowest possible reactor pressure. In view of this, in the following we have considered reactor temperature as the only control variable. While computing, we assumed that the flashing of EG,W and A was occurring at the end of small discrete time intervals with polymerization occurring during this time interval. To be able to determine the concentrations of volatile components in the reaction mass, a vapour-liquid equilibrium of Raoult's law type is used. A suitable Hamiltonian has been defined and the resulting state and adjoint equations have been solved using fourth-order Runge-Kutta method. Using the

the control vector iteration method for first variation
technique, optimal temperature profile has been determined.
For the second variation method, the initial guess temperature
should be very close to the optimal results; otherwise there
is no convergence. Also the first order technique becomes
very slow, especially as the optimum is approached. In view
of this, we have adopted the combined method in which we switch
from the gradient method to the second variation technique
when the convergence becomes very slow. Due to nonlinearities
involved in the state and adjoint variable equations and also
due to memory storage space problems, considerable numerical
difficulty is encountered in the optimization procedure. The
storage problem has been overcome by approximating the actual
temperature profile by 100 piecewise continuous curves and
obtaining the intermediate values by linear interpolation. The
results were found not to change when the number of piece-wise
continuous curves was increased to 200, thus justifying the
stability of computations.

CHAPTER 2

FORMULATION

The kinetic model used in this work is given in Table 1
along with the respective rate and equilibrium constants.
The rate constants are found out when zinc acetate catalyst
of 5.6×10^{-4} moles/litre concentration was used. Based on
the reaction mechanism proposed for the polycondensation stage
of PET formation, the mole balance equations for batch reactors
can be written and are given in Table 2. In these equations,
the lower case symbols represent the moles of various functional
groups and V the volume of the reaction mass to be defined later.
The rate constants are functions of temperature and Arrhenius
relation is used for expressing this dependency. The equilibrium
constants are found to be independent of temperature. The
activation energies and the frequency factors of the respective
rate constants along with the equilibrium constants are
summarized in Table 3. In the mole balance equations, Z
represents the total amount of reacted bonds in the reaction
mass. A separate mole balance equation is not written for Z
because it is related to the moles of various functional
groups through stoichiometry. It is given as

$$Z = \frac{1}{2} \left[e_{go} - e_g - e_c - e_v \right] \tag{3}$$

TABLE 1

Reactions Involved in the Polycondensation Stage of PET Formation

Main Reaction

(1) $E_g + E_g \underset{k_3/K_3}{\overset{k_3}{\rightleftarrows}} Z + G$
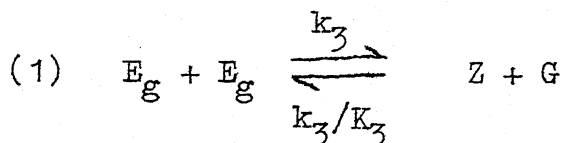
Important Side Reactions

(2) Acetaldehyde Formation

$E_g \xrightarrow{k_4} E_c + A$

(3) Diethylene Glycol Formation

$E_g + G \xrightarrow{k_5} E_c + D$

$E_g + E_g \xrightarrow{k_6} E_c + E_D$

(4) Water formation

$E_c + G \underset{k_7/K_4}{\overset{k_7}{\rightleftarrows}} E_g + w$

$E_c + E_g \underset{k_8/K_5}{\overset{k_8}{\rightleftarrows}} Z + w$
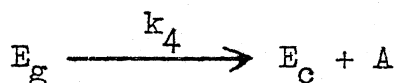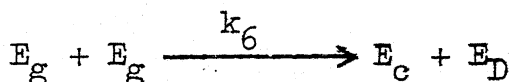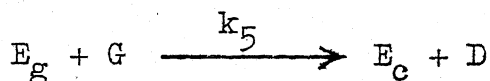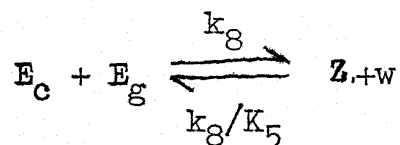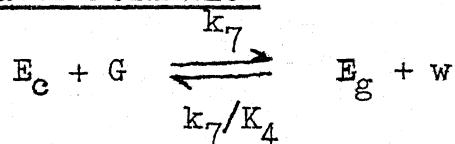
(5) Vinyl Group Formation

$Z \xrightarrow{k_9} E_c + E_v$

$E_g + E_v \xrightarrow{k_3} Z + A$
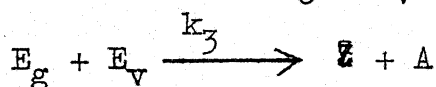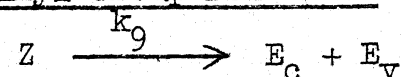
Symbols: $A : CH_3CHO$

$D : OHCH_2 CH_2O CH_2CH_2OH$

Table 1 contd.

E_C : —⟨O⟩—$\overset{O}{\overset{\|}{C}}$ OH

E_D : —⟨O⟩—$\overset{O}{\overset{\|}{C}}OCH_2CH_2OCH_2CH_2OH$

E_g : —⟨O⟩—$\overset{O}{\overset{\|}{C}}OCH_2CH_2OH$

E_V : —⟨O⟩—$\overset{O}{\overset{\|}{C}}OCH = CH_2$

w : H_2O

Z : —⟨O⟩—$\overset{O}{\overset{\|}{C}}OCH_2CH_2O\overset{O}{\overset{\|}{C}}$—⟨O⟩—
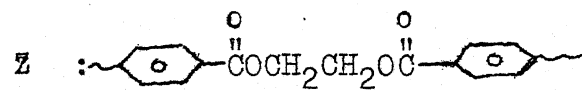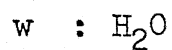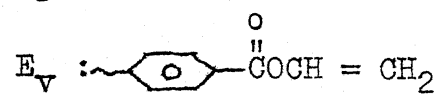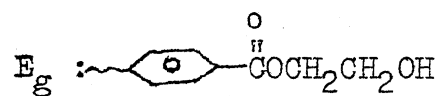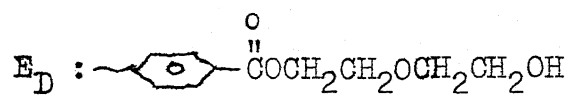
Table 2

Mole Balance Equations for Various Species in Batch
Reactor (State Variable Equations)

(1) $\overset{\circ}{X}_1 = f_1 = \dfrac{de_g}{dt} = V(-2\,R_3-R_4-R_5-2R_6+R_7-R_8-R_{10})$

(2) $\overset{\circ}{X}_2 = f_2 = \dfrac{de_c}{dt} = V(R_4 + R_5 + R_6 - R_7 - R_8 + R_9)$

(3) $\overset{\circ}{X}_3 = f_3 = \dfrac{d\,d}{dt} = V(R_5 + R_6)$

(4) $\overset{\circ}{X}_4 = f_4 = \dfrac{de_v}{dt} = V\,(R_9 - R_{10})$

(5) $\overset{\circ}{X}_5 = f_5 = \dfrac{dg}{dt} = V(R_3-R_5-R_7-Q_{EG})$

(6) $\overset{\circ}{X}_6 = f_6 = \dfrac{dw}{dt} = V(R_7 + R_8 - Q_w)$

(7) $\overset{\circ}{X}_7 = f_7 = \dfrac{da}{dt} = V(R_4 + R_{10} - Q_A)$

$$d = D_{EG} + E_D$$

where

$$R_3 = \frac{k_3}{V^2}\left[e_g{}^2 - \frac{2g}{K_3}(e_{go}-e_g-e_c-e_v)\right]$$

$$R_4 = \frac{k_4}{V}\,e_g$$

$$R_5 = \frac{2k_5}{V^2}\,e_g g$$

$$R_6 = \frac{k_6}{V^2}\,e_g{}^2$$

$$R_7 = \frac{k_7}{V^2}\left(2\,e_c\,g - \frac{e_g\,w}{K_4}\right)$$

$$R_8 = \frac{k_8}{V^2}\left(e_c\,e_g - \frac{w}{K_5}(e_{go}-e_g-e_c-e_v)\right)$$

$$R_9 = \frac{k_9}{2V}\,(e_{go} - e_g - e_c - e_v)$$

$$R_{10} = \frac{k_3}{V^2}\,e_v\,e_g$$

Table 3

Temperature Dependence of Various Rate and Equilibrium Constants

Appearing in Tables 1 and 2

$$k_i = A_i\, e^{-E_i/RT}$$

i	3	4	5	6	7	8	9
Activation energy, E_i Kcal/mole	18.5	29.8	29.8	29.8	17.6	17.6	37.8
Frequency factor A_i lit/mole-min	6.8×10^5	$4.16 \times 10^{7*}$	4.16×10^7	4.16×10^7	1.04×10^6	1.04×10^6	3.6×10^9
Equilibrium constant	$K_3 = 0.5$	—	—	—	$K_4 = 2.5$	$K_5 = 1.25$	—

* Units in min^{-1}

where e_g, e_c, and e_v are the moles of functional groups E_g, E_c, and E_v at time t and e_{go} that of E_g at t=0.

The mole balance equations in Table 2 represents the equations for the state variables X_1, X_2, $X_3, X_4 ... X_7$ and are written in a compact form as

$$\overset{o}{\underline{X}} = \frac{d\underline{X}}{dt} = \underline{f}\,(\underline{X},\, T) \tag{4}$$

where $\underline{X} = [\, X_1,\, X_2,\, X_3,\, X_4,\, X_5,\, X_6,\, X_7\,]^T$ (a)

$\underline{f} = [\, f_1,\, f_2,\, f_3,\, f_4,\, f_5,\, f_6,\, f_7\,]^T$ (b)

(5)

Since the feed to the polycondensation stage is assumed to be BHET alone, the 'initial' conditions for the state variables can be given as

$X_1\,(t=o) = 1.0$ (a)

$X_i\,(t=o) = 0.0, \quad i = 2,3,4,5,6,7$ (b)

(6)

The mole balance equations in Table 2 cannot be solved without knowing the quantities Q_{EG}, Q_W and Q_A, which are present in the mole balance equations for the volatile components, equations (5), (6), and (7) in Table 2. These quantities stand for the amounts of ethylene glycol, water, and acetaldehyde flashed from the reaction mass due to the boiling phenomena occurring at the temperature and pressure applied on it. In order to determine these unknown quantities, a sound theoretical

model should be developed which relates the amounts flashed
to the conditions existing in the reactor i.e., to the
temperature, pressure, and composition of the liquid boiling.
Here, we proposed a simplified model for the flashing process
in which it is assumed that the flashing is occurring at the
end of discrete times whereas the polymerization occurs
without flashing during this time interval. This means that
Q_{EG}, Q_W, and Q_A should be replaced by $Q_{EG} \delta(t - n\Delta t)$,
$Q_W \delta(t - n \Delta t)$ and $Q_A \delta(t - n \Delta t)$ where 'n' is the number
of discrete intervals in time, t, Δt is the width of the
time interval and δ is the dirac delta function. Using the
above model, we carried out the computations hoping that the
results obtained would simulate the flashing process.

Since the polymer molecules are assumed to be involatile,
the total pressure in the reactor should be equal to the sum of
the partial pressures exerted by EG, W, and A. Assuming
that the vapour leaving the boiling liquid is in thermodynamic
equilibrium with the residual liquid and assuming Raoult's
law is applicable, we can write.

$$P_T = P_G^O X_G + P_W^O X_W + P_A^O X_A \qquad (7)$$

where P_T is the total pressure in the reactor, P_G^O, P_W^O, and
P_A^O are the pure-component vapour pressures of EG, W, and A
respectively at the temperature existing in the reactor and X_G,
X_W and X_A are the corresponding mole fractions of the components

in the liquid phase. Usually the temperature of the reaction
mass is greater than the critical temperature of acetaldehyde and
therefore it cannot exist in the reaction mass. X_G, X_W and X_A
in the reaction mass are obtained from the following equations

$$X_G = \frac{g}{\frac{1}{2}(e_g + e_c + d + e_v) + g + w + a} \tag{a}$$

$$X_W = \frac{w}{\frac{1}{2}(e_g + e_c + d + e_v) + g + w + a} \tag{b}$$

$$X_A = \frac{a}{\frac{1}{2}(e_g + e_c + d + e_v) + g + w + a} \tag{c}$$

$$\tag{8}$$

where the first term in the denominator in the above equations
represents the moles of polymer, M_p. Since the vapour
pressures of EG, W, and A can be obtained from any standard book
on thermodynamics,[35] X_G, X_W, and X_A in the reaction mass can
easily be computed.

The volume of the reaction mass, V, is computed using
the relation

$$V = V_P\,M_P + V_G X_G + V_W\,X_W \tag{9}$$

where V_P, V_G, and V_W are the molar volumes of the polymer,
ethylene glycol, and water respectively. V_P is assumed to be
the molar volume of DMT. V_P, V_G and V_W are temperature dependent
and the equations expressing this dependency are given later.

Since our main goal is to determine the optimal temperature

profile in the reactor, we should select some method of optimization. In this work, we applied the variational method of optimization to PET reactor. We assumed the following functional relationship for the volatile components, EG,W and A:

$$g = g(T, \quad e_g, \quad e_c, \quad d, \quad e_v) \qquad (a)$$

$$w = w(T, \quad e_g, \quad e_c, \quad d, \quad e_v) \qquad (b)$$

$$a = a(T, \quad e_g, \quad e_c, \quad d, \quad e_v) \qquad (c)$$

$$(10)$$

and defined the state variable vector as in Eq.5(a).

Objective Function

As mentioned earlier, the quality of the polymer produced is largely dependent on the various side products formed. Since the reaction time and the pressure inside the reactor are specified, the only variable that can be controlled is temperature. The temperature should be controlled in such a way that the number average molecular weight of the polymer produced reaches a desired value, μ_{nd}, in the shortest time possible and at the same time minimizing the amount of side products formed. It is also found that an excess of diethylene glycol, e_D, in the final product is undesirable because it lowers the melting point of the fibre, but to improve the dyeing ability of the polymer, some small amount of it (say D*) is needed. Bearing all these objectives in mind, the objective function is written as

$$I \{T(t)\} = \alpha_1 (e_D - D^*)]_{t=t_f} + \int_o^{t_f} \{ \frac{\alpha_2}{\mu_{nd}^2} (\frac{e_{go} - \mu_{nd}^*}{e_g})^2$$

$$+ \alpha_3 (e_c^2 + e_v^2) \} dt \tag{11}$$

where $T(t)$ is the temperature as a function of time in a
batch reactor to be determined optimally so that I is minimized,
t_f is the reaction time, μ_{nd} is the desired value of the
number average molecular weight of the polymer, D* is the
desired amount of diethylene glycol in the polymer and α_1, α_2
and α_3 are the weighting parameters representing the relative
importance of diethylene glycol in the product, number average
molecular weight of the product and the amount of side products
formed. It is found that the weighting parameters have a
considerable influence on the optimal temperature profile to
be determined. The objective function proposed allows more
flexibility in reactor design and forces the product from a
batch reactor to be as close to the desired value, μ_{nd}, as
possible. This is because at the design stage, a few percent
deviation in the value of μ_{nd} can be tolerated without
causing serious damage to the physical properties of the
polymer.

CHAPTER 3

METHOD OF SOLUTION

With the definition of the state variable vector in
Eq.5(a) and the objective function in Eq.(11), the optimal
temperature profiles in a batch reactor can be easily obtained.
One defines the Hamiltonian, H as

$$H = \frac{\alpha_2}{\mu_{nd}^2} \left(\frac{e_{go}}{e_g} - \mu_{nd} \right)^2 + \alpha_3 (e_c^2 + e_v^2) + \sum_{i=1}^{4} \lambda_i f_i \tag{12}$$

where λ_i are the adjoint variables given by

$$\frac{-d\lambda_i}{dt} = \frac{\partial H}{\partial X_i}, \quad i = 1,2,3,4 \tag{13}$$

with the final conditions as

$$\lambda_i(t_f) = \frac{\partial}{\partial X_i} \left[\alpha_i (e_D - D*) \right]_{t=t_f}, \quad i = 1,2,3,4 \tag{14}$$

The detailed equations for the adjoint variables along with
the final conditions are given in Table 4.

Having solved the state variable equations in Table 2, and
the adjoint variable equations, Eq.(13), the optimal temperature
profile is obtained by using the necessary condition.

$$\frac{\partial H}{\partial T} = 0 = \sum_{i=1}^{4} \lambda_i \frac{\partial f_i (X_1, X_2, X_3, X_4, T)}{\partial T} \tag{15}$$

The detailed equations for $\frac{\partial f_i}{\partial T}$ are given in Table 5.

In obtaining the optimal temperature profiles, the temperature is constrained to lie between two limiting values

$$400°K \leq T \leq 600°K \tag{16}$$

and it can be shown that the optimality relations described earlier can be used for this constrained problem.

In order to obtain the optimal temperature profiles, use can be made of several iterative procedures existing in the literature. In all those iterative procedures, at least one set of equations from the governing equations for the optimization zations from the governing equations for the optimization problem should be assumed. For example, this can be the unknown initial conditions on the adjoint variables or the entire set of control variables. With the assumed values of any of these variables, the equations governing the problem are solved and later, the original assumption is corrected by using some technique. The new values calculated yield a result closer to the optimal values than the assumed values. This procedure is repeated assuming convergence occurs. In this way at least a local optimum can be found out.

Dynamic programming can be used as a method for the calculation of optimal temperature profile. But this method has a drawback in that it needs a considerable large memory

storage space for solving system equations in which there
are only three or more state variables. Despite this drawback,
it can be used for systems in which there are constraints on
control and state variables. Another method called the
"gradient method" or "control-vector **iteration** procedure"
as suggested by Ray and Szekely[36] can be used. In this method
the system equations and the boundary conditions are fixed
and an iterative procedure developed which converges to the
desired performance index.

We have chosen the control vector iteration procedure
for computing optimum temperature profiles. Moreover, the
algorithm for this method is simple to program for a computer
But the disadvantage with this method is that it becomes very
slow especially when the optimum is approached and does not
converge to the true optimum. At this stage the computations
should be stopped and some other method must be used for
using the convergence or for finding the true optimum. As
suggested by Lapidus and Luus[37], we used the second variation
method for achieving the **true** optimum. But this method is found
to lead to a lot of instability problems if an incorrect
initial guess is used. In order to overcome this problem, we
used a combined method for solving our optimization problem.
In this, gradient method is used for the first few iterations
or till the point where the change in the value of the
objective function from iteration to iteration is small and

at this stage, one switches over to the second variation method until the optimal profile is obtained. We found that the combined method works better than ~~the~~ each method applied separately. These two methods are described separately below:

Control Vector Iteration Procedure:

In this method (sometimes called the gradient method), one assumes a temperature profile $T_0(t)$. With this the state variable equations (Table 2) are integrated in the forward direction (from $t=0$ to t_f), storing the values at small time intervals. The objective function is computed using these stored values of state variables. The adjoint variable equations (Table 4) are integrated in the backward direction from $t=t_f$ to 0) and finally the assumed temperature profile is corrected using the equation

$$T^{new}(t) = T^{old}(t) - \varepsilon \frac{\partial H}{\partial T} \qquad (17)$$

where ε is a parameter which determines the magnitude of the correction to be employed in the old temperature profile and is assumed to be independent of time t. Assuming a value of ε, the temperature profile is corrected and the above procedure is repeated till the necessary condition given in Eq.(15) is satisfied. It is important to note that the value of $\frac{\partial H}{\partial T}$ in Eq.(15) in the first iteration will not necessarily be zero since the assumed temperature profile is not optimal. This

method is extremely efficient and the optimal temperature profile is usually obtained after two or three iterations.

In the literature, there exist several techniques for finding the value of ε from iteration to iteration. In this work, we selected that technique in which an interactive computer terminal is used. In this technique, various values of ε are assumed and the new temperature profiles are obtained using Eq.(17) with the temperature profiles obtained, the integration of state variable equations is carried out and the corresponding objective functions are calculated. The various values of ε and the corresponding objective functions are examined on the terminal and the approximate value of ε corresponding to the lowest value of the objective function is selected as the optimum ε for the next iteration of calculations. With the value of optimum ε, the old temperature profile is corrected and the entire procedure is repeated till the value of the objective function does not change significantly from iteration to iteration. The technique suggested by Denn[38], does not require an interactive computer terminal, but there are several situations when this leads to unstable results. We selected the one requiring since it yields the optimum ε in a minimum number of trials.

Integration of State Variable and Adjoint Variable Equations:

The integration of these equations is carried out using fourth-order Runge-Kutta method. As is evident, the integration of the state variable equations (Table 2) is not possible since the quantities Q_{EG}, Q_W, and Q_A are unknown. However, to overcome this problem, we used the method proposed earlier in this thesis. In this method, for a given time interval, the integration of the mole balance equations for the volatile components is carried out without the terms Q_{EG}, Q_W, and Q_A and then at the end of this time interval a flash subroutine is used to find the concentrations of ethylene glycol and water such that the Eq.(7) remains satisfied. In the flash subroutine, with the values of the state variables obtained after integration time interval Δt, the total pressure P_T is computed using Eq.(7) and if the computed value is found to be greater than the actual pressure applied in the reactor, then it is obvious that some amount of flashing has occurred. The amount flashed can be found out using equilibrium flash calculations. If the computed pressure is found to be less than the actual pressure, then there is no flashing. This way, the state variable equations are integrated for the entire time of polymerization. The total time of polymerization has been assumed to be two hours and this time has been divided into 2000 equal intervals in order to avoid excessive memory storage

space. But when the computations are carried out with a Δt of 1×10^{-3} hr, a numerical instability was encountered, because of the fast changes in the concentration of EG. It is also found to become negative with $\Delta t = 10^{-3}$ hr, when the vigorous flashing of ethylene glycol starts at lower pressures. We found that by decreasing Δt to 2×10^{-4} hr, numerically stable results could be obtained. But when Δt is reduced to this value, there is a problem of memory storage. However, we have overcome this problem, by storing the values of state variables at every time interval of 10^{-3} hr or at 2001 points of time. The integration of the adjoint variable equations (Table 4) is carried out with Δt of 2×10^{-4} hr, using linear interpolation for the intermediate values of state variables. The adjoint variables are also stored at 2001 points of time.

For all the above integrations to proceed, some initial guess should be used for the temperature profile. Here we assumed that the temperature profile is isothermal and approximated the actual profile by 100 piece-wise continuous curves. In the integrations, the intermediate values of temperature are simply found by linear interpolation.

A listing of the computer program for the control vector iteration method is provided in Appendix.

Table 4

Equations Governing the Adjoint Variables

$$\frac{d\lambda_1}{dt} = \frac{2\alpha_2 \, e_{go}}{\mu_{nd}^2 \, e_g^2} \left(\frac{e_{go}}{e_g} - \mu_{nd}\right) - \left(\frac{\partial f_1}{\partial e_g}\right) \lambda_1 + \left(\frac{\partial f_2}{\partial e_g}\right) \lambda_2$$

$$+ \left(\frac{\partial f_3}{\partial e_g}\right) \lambda_3 + \left(\frac{\partial f_4}{\partial e_g}\right) \lambda_4$$

$$\frac{d\lambda_2}{dt} = -2\alpha_3 \, e_c - \left(\frac{\partial f_1}{\partial e_c}\right) \lambda_1 + \left(\frac{\partial f_2}{\partial e_c}\right) \lambda_2 + \left(\frac{\partial f_3}{\partial e_c}\right) \lambda_3$$

$$+ \left(\frac{\partial f_4}{\partial e_c}\right) \lambda_4$$

$$\frac{d\lambda_3}{dt} = - \left(\frac{\partial f_1}{\partial d}\right) \lambda_1 + \left(\frac{\partial f_2}{\partial d}\right) \lambda_2 + \left(\frac{\partial f_3}{\partial d}\right) \lambda_3 + \left(\frac{\partial f_4}{\partial d}\right) \lambda_4$$

$$\frac{d\lambda_4}{dt} = -2\alpha_3 \, e_v - \left(\frac{\partial f_1}{\partial e_v}\right) \lambda_1 + \left(\frac{\partial f_2}{\partial e_v}\right) \lambda_2 + \left(\frac{\partial f_3}{\partial e_v}\right) \lambda_3 + \left(\frac{\partial f_4}{\partial e_v}\right) \lambda_4$$

where

$$\frac{\partial f_1}{\partial e_g} = \frac{-4 k_3}{V} \left[e_g + \frac{g}{K_3} - \frac{(e_{go} - e_g - e_c - e_v)}{K_3} \frac{\partial g}{\partial e_g} \right]$$

$$-k_4 - \frac{2 k_5}{V} \left[e_g \frac{\partial g}{\partial e_g} + g \right] - \frac{4 k_6 e_g}{V}$$

$$+ \frac{k_7}{V} \left[2 e_c \frac{\partial g}{\partial e_g} - \frac{e_g}{K_4} \frac{\partial w}{\partial e_g} - \frac{w}{K_4} \right]$$

$$- \frac{k_8}{V} \left[e_c + \frac{w}{K_5} - \frac{(e_{go} - e_g - e_c - e_v)}{K_5} \frac{\partial w}{\partial e_g} \right]$$

$$- \frac{k_3 \, e_v}{V}$$

$$\frac{\partial f_2}{\partial e_g} = k_4 + \frac{2\,k_5}{V}(e_g\,\frac{\partial g}{\partial e_g} + g) + \frac{2\,k_6\,e_g}{V}$$

$$+ \frac{k_7}{V}(-2\,e_c\,\frac{\partial g}{\partial e_g} + \frac{e_g}{K_4}\,\frac{\partial w}{\partial e_g} + \frac{w}{K_4})$$

$$- \frac{k_8}{V}(e_c + \frac{w}{K_5} - \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\,\frac{\partial w}{\partial e_g})$$

$$- \frac{k_9}{2}$$

$$\frac{\partial f_3}{\partial e_g} = 2\,\frac{k_5}{V}(e_g\,\frac{\partial g}{\partial e_g} + g) + \frac{2\,k_6\,e_g}{V}$$

$$\frac{\partial f_4}{\partial e_g} = -\frac{k_9}{2} - \frac{k_3\,e_v}{V}$$

$$\frac{\partial f_1}{\partial e_c} = \frac{4\,k_3}{V\,K_3}(-g + (e_{go}-e_g-e_c-e_v)\,\frac{\partial g}{\partial e_c})$$

$$- \frac{2\,k_5\,e_g}{V}\,\frac{\partial g}{\partial e_c} + \frac{k_7}{V}(2e_c\,\frac{\partial g}{\partial e_c} + 2g -$$

$$\frac{e_g}{K_4}\,\frac{\partial w}{\partial e_c}) - \frac{k_8}{V}(e_g + \frac{w}{K_5} - \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\,\frac{\partial w}{\partial e_c},$$

$$\frac{\partial f_2}{\partial e_c} = \frac{2k_5\,e_g}{V}\,\frac{\partial g}{\partial e_c} - \frac{k_7}{V}(2\,e_c\,\frac{\partial g}{\partial e_c} + 2g - \frac{e_g}{K_4}\,\frac{\partial w}{\partial e_c})$$

$$- \frac{k_8}{V}(e_g + \frac{w}{K_5} - \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\,\frac{\partial w}{\partial e_c})$$

$$- \frac{k_9}{2V}$$

$$\frac{\partial f_3}{\partial e_c} = \frac{2\,k_5\,e_g}{V}\,\frac{\partial g}{\partial e_c}$$

$$\frac{\partial f_4}{\partial e_c} = -\frac{k_9}{2}$$

$$\frac{\partial f_1}{\partial d} = \frac{\partial f_2}{\partial d} = \frac{\partial f_3}{\partial d} = \frac{\partial f_4}{\partial d} = 0.0$$

$$\frac{\partial f_1}{\partial e_v} = \frac{4 \, k_3}{v \, k_3} (-g + (e_{go} - e_g - e_c - e_v) \frac{\partial g}{\partial e_v})$$

$$- \frac{2 \, k_5 \, e_g}{v} \frac{\partial g}{\partial e_v} + \frac{k_7}{v} (2 \, e_c \frac{\partial g}{\partial e_v} - \frac{e_g}{K_4} \frac{\partial w}{\partial e_v})$$

$$- \frac{k_8}{v \, K_5} (-w + (e_{go} - e_g - e_c - e_v) \frac{\partial w}{\partial e_v})$$

$$- \frac{k_3 \, e_g}{v}$$

$$\frac{\partial f_2}{\partial e_v} = \frac{2 \, k_5 \, e_g}{v} \frac{\partial g}{\partial e_v} + \frac{k_7}{v} (-2 e_c \frac{\partial g}{\partial e_v} + \frac{e_g}{K_4} \frac{\partial w}{\partial e_v}$$

$$+ \frac{k_8}{v K_5} (-w + (e_{g \bullet} - e_g - e_c - e_v) \frac{\partial w}{\partial e_v}) - \frac{k_9}{2}$$

$$\frac{\partial f_3}{\partial e_v} = \frac{2 \, k_5 \, e_g}{v} \frac{\partial g}{\partial e_v}$$

$$\frac{\partial f_4}{\partial e_v} = - \frac{k_9}{2} - \frac{k_3 \, e_g}{v}$$

$$\frac{\partial g}{\partial e_g} = \frac{\partial g}{\partial e_c} = \frac{\partial g}{\partial d} = \frac{\partial g}{\partial e_v} = \frac{P_T}{2(P_G^o - P_T)}$$

$$\frac{\partial w}{\partial e_g} = \frac{\partial w}{\partial e_c} \quad \frac{\partial w}{\partial d} \quad \frac{\partial w}{\partial e_v} = \frac{P_T}{2(P_W^o - P_T)}$$

Final Conditions on the Adjoint Variables

$$\lambda_1 (t_f) = 0$$

$$\lambda_2 (t_f) = 0$$

$$\lambda_3 (t_f) = \alpha_1$$

$$\lambda_4 (t_f) = 0$$

TABLE 5

DETAILED EQUATIONS FOR THE TERMS $\left(\dfrac{\partial f_i}{\partial T}\right)$

$$\frac{\partial f_1}{\partial T} = -2\,e_g^2\,\frac{d(k_3/V)}{dT} + \frac{4\,(e_{go} - e_g - e_c - e_v)}{K_3}\left[\left(\frac{k_3}{V}\right)\frac{\partial g}{\partial T}\right.$$

$$+ g\,\frac{d(k_3/V)}{dT}\Big] - \frac{d(k_4)}{dT}\,e_g - 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial g}{\partial T}\right.$$

$$+ g\,\frac{d(k_5/V)}{dT}\Big] -2\,e_g^2\,\frac{d(k_6/V)}{dT} + 2\,e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial g}{\partial T}\right.$$

$$+ g\,\frac{d(k_7/V)}{dT}\Big] - \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial W}{\partial T} + w\,\frac{d(k_7/V)}{dT}\right]$$

$$- e_c\,e_g\,\frac{d(k_8/V)}{dT} + \frac{(e_{go} - e_g - e_c - e_v)}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial w}{\partial T}\right.$$

$$+ w\,\frac{d(k_8/V)}{dT}\Big] - e_v\,e_g\,\frac{d(k_3/V)}{dT}$$

$$\frac{\partial f_2}{\partial T} = e_g\,\frac{d(k_4)}{dT} + 2\,e_g\left[\left(\frac{k_5}{V}\right)g + g\,\frac{d(k_5/V)}{dT}\right]$$

$$- e_g^2\,\frac{d(k_6/V)}{dT} - 2\,e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial g}{\partial T} +\right.$$

$$g\,\frac{d(k_7/V)}{dT}\Big] + \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial w}{\partial T} + w\,\frac{d(k_7/V)}{dT}\right]$$

$$-e_c\,e_g\,\frac{d(k_8/V)}{dT} + \frac{(e_{go} - e_g - e_c - e_v)}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial w}{\partial T}\right.$$

$$+ w\,\frac{d(k_8/V)}{dT}\Big] + \frac{(e_{go} - e_g - e_c - e_v)}{2}\,\frac{d(k_9)}{dT}$$

$$\frac{\partial f_3}{\partial T} = 2\, e_g \left[\left(\frac{k_5}{V}\right) \frac{\partial g}{\partial T} + g\, \frac{d(k_5/V)}{dT} \right]$$

$$+ e_g^2 \, \frac{d(k_6/V)}{dT}$$

$$\frac{\partial f_4}{\partial T} = \frac{(e_{go} - e_g - e_c - e_v)}{2} \, \frac{d(k_9)}{dT} - e_v\, e_g \, \frac{d(k_3/V)}{dT}$$

where

$$\frac{d(k_i/V)}{dT} = \frac{A_i\, e^{-\frac{E_i}{RT}}\, E_i}{VRT^2} - \frac{A_i\, e^{-\frac{E_i}{RT}}}{V^2}\, \frac{dV}{dT} \quad,$$

$$\frac{d(k_i/V)}{dT} = \frac{A_i\, E_i\, e^{-\frac{E_i}{RT}}}{RT^2} \quad, \qquad i = 3,5,6,7,8$$

$$\qquad\qquad\qquad\qquad\qquad\qquad i = 4,9$$

$$\frac{dV}{dT} = \left(\frac{dV_P}{dT}\right) M_P + V_G\, \frac{d\, X_G}{dT} + X_G \left(\frac{d\, V_G}{dT}\right) +$$

$$V_W \left(\frac{d\, X_W}{dT}\right) + X_W \left(\frac{d\, V_W}{dT}\right)$$

where

$$V_P = 191.5\,(1.0 + 0.0014\,(T-413))/1000$$

$$V_G = 60.6\,(1.0 + 0.0014\,(T-413))/1000$$

$$V_W = (19.422 + 0.025\,(T-413))/1000$$

$$\frac{dV_P}{dT} = 2.68 \times 10^{-4}; \qquad \frac{dV_G}{dT} = 8.48 \times 10^{-5};$$

$$\frac{dV_W}{dT} = 2.5 \times 10^{-5}$$

$$\frac{dX_G}{dT} = \frac{\{(\frac{e_g + e_c + d + e_v}{2}) + w + a\} \frac{\partial g}{\partial T}}{\{(\frac{e_g + e_c + d + e_v}{2}) + g + w + a\}^2}$$

$$\frac{dX_W}{dT} = \frac{\{(\frac{e_g + e_c + d + e_v}{2}) + g + a\} \frac{\partial w}{\partial T}}{\{(\frac{e_g + e_c + d + e_v}{2}) + g + w + a\}^2}$$

$$\frac{\partial g}{\partial T} = \frac{-[g \frac{dP_G^\circ}{dT} + w \frac{dP_W^\circ}{dT}]}{(P_G^\circ - P_T)}$$

$$\frac{\partial w}{\partial T} = \frac{-[g \frac{dP_G^\circ}{dT} + w \frac{dP_W^\circ}{dT}]}{(P_W^\circ - P_T)}$$

$$\frac{dP_G^\circ}{dT} = P_G^\circ \left[\frac{3729}{T^2} - \frac{4.042}{T} \right]$$

$$\frac{dP_W^\circ}{dT} = P_W^\circ \left[\frac{1757.853}{(T-33.274)^2} \right]$$

$$P_G^\circ = \{ 10^{[8.064103 - \frac{1757.853}{(T-33.274)}]} \} /760$$

$$P_W^\circ = \{ 10^{[21.61 - \frac{3729}{T} - 4.042 \log T]} \} /760$$

Second Variation Method:

As mentioned earlier, this method leads to a lot of numerical instability problems if an incorrect initial guess for the temperature profile is used. In order to overcome these problems, control vector **iteration** method is used for the first few iterations and with the near optimal temperature profile obtained using this method, the present method isstarted.

In this method, with the near optimal temperature profile obtained from the control vector iteration method, the state-variable equations (Table 2) are integrated in the forward direction (from $t=0$ to t_f) storing the values of these at every time interval of 1×10^{-3} hr. The objective function given by Eq.(11) is computed using the stored values of state variables and the adjoint variable equations (Table 4) are integrated in the backward direction (from $t = t_f$ to 0). In addition to these, the following equations are integrated in the backward direction (from $t = t_f$ to 0),

$$\overset{\circ}{\underline{P}} = \frac{d\underline{P}}{dt} = -\left(\frac{\partial^2 H}{\partial \underline{X}^2}\right) - \left[\underline{P}\left(\frac{\partial \underline{f}^T}{\partial \underline{X}}\right)^T + \left(\frac{\partial \underline{f}^T}{\partial \underline{X}}\right)\underline{P}\right] + \underline{K}\,\underline{R} \qquad (18)$$

$$\underline{P} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{bmatrix} \quad (a)$$

$$\overset{\circ}{\underline{P}} = \begin{bmatrix} \frac{dP_{11}}{dt} & \frac{dP_{12}}{dt} & \frac{dP_{13}}{dt} & \frac{dP_{14}}{dt} \\ \frac{dP_{21}}{dt} & \frac{dP_{22}}{dt} & \frac{dP_{23}}{dt} & \frac{dP_{24}}{dt} \\ \frac{dP_{31}}{dt} & \frac{dP_{32}}{dt} & \frac{dP_{33}}{dt} & \frac{dP_{34}}{dt} \\ \frac{dP_{41}}{dt} & \frac{dP_{42}}{dt} & \frac{dP_{43}}{dt} & \frac{dP_{44}}{dt} \end{bmatrix} \quad (b)$$

$$(19)$$

$$\underline{X} = [X_1, X_2, X_3, X_4]^T \qquad (a)$$

$$\underline{f} = [f_1, f_2, f_3, f_4]^T \qquad (b)$$

(20)

$$\underline{R} = (\frac{\partial^2 H}{\partial T \partial \underline{X}}) + (\frac{\partial \underline{f}^T}{\partial T}) \underline{P} \qquad (21)$$

$$\underline{K} = \underline{R}^T (\frac{\partial^2 H}{\partial T^2})^{-1} \qquad (22)$$

$$\frac{\partial^2 H}{\partial T \partial \underline{X}} = \frac{\partial^2 J}{\partial T \partial \underline{X}} + \sum_{i=1}^{4} \lambda_i \frac{\partial^2 f_i}{\partial T \partial \underline{X}} \qquad (23)$$

$$\frac{\partial^2 H}{\partial T^2} = \sum_{i=1}^{4} \lambda_i \frac{\partial^2 f_i (X_1, X_2, X_3, X_4, T)}{\partial T^2} \qquad (24)$$

$$\frac{\partial \underline{f}^T}{\partial T} = [\frac{\partial f_1}{\partial T}, \frac{\partial f_2}{\partial T}, \frac{\partial f_3}{\partial T}, \frac{\partial f_4}{\partial T}] \qquad (25)$$

$$\frac{\partial^2 H}{\partial \underline{X}^2} = \frac{\partial^2 J}{\partial \underline{X}^2} + \sum_{i=1}^{4} \lambda_i \frac{\partial^2 f_i}{\partial \underline{X}^2} \qquad (26)$$

$$\overset{o}{\underline{q}}{}^{\emptyset} = \frac{d\underline{q}}{dt} = \underline{R}^T (\frac{\partial^2 H}{\partial T^2})^{-1} (\frac{\partial H}{\partial T}) + \underline{R}^T (\frac{\partial^2 H}{\partial T^2})^{-1} (\frac{\partial \underline{f}^T}{\partial T}) \underline{q}$$

$$-(\frac{\partial \underline{f}^T}{\partial \underline{X}}) \underline{q} \qquad (27)$$

where $\underline{q} = [q_{11}, q_{21}, q_{31}, q_{41}]^T \qquad (a)$

$$\overset{o}{\underline{q}}{}^{\emptyset} = [\frac{dq_{11}}{dt}, \frac{dq_{21}}{dt}, \frac{dq_{31}}{dt}, \frac{dq_{41}}{dt}]^T \qquad (b)$$

(28)

$$\frac{\partial H}{\partial T} = \sum_{i=1}^{4} \lambda_i \frac{\partial f_i (X_1, X_2, X_3, X_4, T)}{\partial T} \qquad (29)$$

The detailed equations for the various terms, $\dfrac{\partial^2 J}{\partial \underline{X}^2}$,

$\dfrac{\partial^2 f_i}{\partial \underline{X}^2}$, $\dfrac{\partial^2 J}{\partial T \partial \underline{X}}$, $\dfrac{\partial^2 f_i}{\partial T \partial \underline{X}}$, and $\dfrac{\partial^2 f_i}{\partial T^2}$, involved in the

equations (18)-(29) are given in Tables 6, 7, and 8 respectively.
The equations for $\dfrac{\partial f_i}{\partial T}$ are already given in Table 5.

In Eq.(18)

$$\frac{\partial \underline{f}^T}{\partial \underline{X}} = \begin{bmatrix} \dfrac{\partial f_1}{\partial e_g} & \dfrac{\partial f_2}{\partial e_g} & \dfrac{\partial f_3}{\partial e_g} & \dfrac{\partial f_4}{\partial e_g} \\[2ex] \dfrac{\partial f_1}{\partial e_c} & \dfrac{\partial f_2}{\partial e_c} & \dfrac{\partial f_3}{\partial e_c} & \dfrac{\partial f_4}{\partial e_c} \\[2ex] \dfrac{\partial f_1}{\partial d} & \dfrac{\partial f_2}{\partial d} & \dfrac{\partial f_3}{\partial d} & \dfrac{\partial f_4}{\partial d} \\[2ex] \dfrac{\partial f_1}{\partial e_v} & \dfrac{\partial f_2}{\partial e_v} & \dfrac{\partial f_3}{\partial e_v} & \dfrac{\partial f_4}{\partial e_v} \end{bmatrix} \qquad (30)$$

The detailed equations for the elements involved in the above
matrix are included in Table 4.

Equation (18) is also known as the first-order Riccati equation.
In this $\underline{P}$ is a symmetric matrix of the order (4x4) as given in
Eq.19(a). It is also related to the adjoint variables by the
equation

$$\underline{P} = \left(\frac{\partial \underline{\lambda}^T}{\partial \underline{X}} \right)^T \qquad (31)$$

The final conditions on $\underline{P}$ are:

$$\underline{P}(t_f) = \frac{\partial^2}{\partial \underline{X}^2} \left[\alpha_1 (e_D - D*) \right] \tag{32}$$

In Eq.(27), $\underline{q}$ is a vector function of the order (4x1) given by Eq.28(a). The final conditions on q are

$$\underline{q}(t_f) = \underline{0} \tag{33}$$

During the backward integration of the equations (18) and (27), the values of $\underline{P}$, $\underline{q}$, $\underline{R}$, $\dfrac{\partial f_i}{\partial T}$, $\dfrac{\partial H}{\partial T}$, and $\dfrac{\partial^2 H}{\partial T^2}$ are stored in the computer memory. Using these stored values, the perturbation equations for the state variables given by

$$\delta \overset{\circ}{\underline{X}} = \frac{d(\delta \underline{X})}{dt} = \left(\frac{\partial \underline{f}^T}{\partial \underline{X}}\right) \delta \underline{X} + \left(\frac{\partial \underline{f}^T}{\partial T}\right) \delta T \tag{34}$$

where

$$\delta \underline{X} = \left[\delta X_1, \ \delta X_2, \ \delta X_3, \ \delta X_4 \right]^T \qquad \text{(a)}$$

$$\delta \overset{\circ}{\underline{X}} = \left[\frac{d \, \delta X_1}{dt}, \ \frac{d \, \delta X_2}{dt}, \ \frac{d \, \delta X_3}{dt}, \ \frac{d \, \delta X_4}{dt} \right]^T \qquad \text{(b)} \tag{35}$$

are integrated in the forward direction (from $t=0$ to t_f) with the initial conditions as

$$\delta \underline{X}(t=0) = \underline{0} \tag{36}$$

The above initial conditions are obtained noting that the initial conditions on the state variables given by Eq.(6) are constant.

The values of $\delta \underline{X}$ are also stored in the computer memory. With the stored values of $\underline{P}$, $\underline{q}$, $\underline{R}$, $\dfrac{\partial f_i}{\partial T}$, $\left(\dfrac{\partial H}{\partial T}\right)$, $\left(\dfrac{\partial^2 H}{\partial T^2}\right)$, and $\delta \underline{X}$, the temperature profile is corrected using the equation

$$T^{(j+1)}(t) = T^{(j)}(t) - \left[\varepsilon \left(\frac{\partial^2 \bar{H}}{\partial T^2} \right)^{-1} \left(\frac{\partial \bar{H}}{\partial T} + \frac{\partial f^T}{\partial T} \underline{q} \right) \right]^{(j)}$$

$$- \left[\left(\frac{\partial^2 \bar{H}}{\partial T^2} \right)^{-1} \underline{R} \right]^{(j)} \delta \underline{X}^{(j)} \qquad (37)$$

where 'j' stands for the iteration number and ε is a step
size parameter determining the magnitude of the correction
employed in each iteration. This parameter is introduced by
Merriam[39] to prevent overstepping in the temperature correction
and also to maintain the validity of the approximations used
in deriving the Eq.(37). According to Merriam, the limits set for
ε is $0 < \varepsilon \leq 1$. The value of ε is found out using an inter-
active computer terminal. For the first iteration, the value
of ε is taken to be one and with the temperature profile
obtained from Eq.(37), the state variable equations are integrated
and simultaneously the objective function is computed using
Eq.(11). If this objective function is found to be greater than
the previous one, the value of ε is halved and the iteration
is continued. This procedure is repeated till the objective
function decreases. The value of ε corresponding to this
objective function is taken to be the optimum ε and the
temperature profile is corrected. This completes one iteration
of computation. The entire procedure is repeated till the
objective function does not change significiantly from iteration
to iteration. It is found that with a value of $\varepsilon = 1.0$, faster
convergence is attained i.e. the lowest value of objective

function is obtained and if $\varepsilon > 1.0$, there is no convergence.

The optimal temperature profile is obtained using the necessary condition given by Eq.(15) and the sufficient conditions given by the equations

$$\frac{\partial^2 H}{\partial T^2} > 0 \qquad \text{for} \quad 0 \leq t \leq t_f \qquad \text{(a)}$$

$$\underline{P}(t) \text{ finite for } 0 \leq t \leq t_f \qquad \text{(b)}$$

$$(38)$$

According to Bryson and Ho[40], the above sufficient conditions are known as (i) the convexity condition (or strengthened Legendre-Clebsch condition), and (ii) the condition that no conjugate points **exist** on the path (or Jacobi condition). These two sufficient conditions determine whether any neighbouring extremal paths exist. If any of these sufficient conditions is not satisfied , then the second variation method will diverge which in turn implies that the neighbouring extremal paths does not exist. If $\underline{P}(t) \to \infty$ at t=t', where $0 \leq t' < t_f$, the integration of Eq.(20) should be stopped because there is a conjugate path for t< t'. This behaviour of $\underline{P}$, we encountered in our problem and as suggested by Bryson and Ho, we stopped the integration at this point and assumed that the temperature profile for t≤ t' does not change or it is the optimal profile that can be obtained under the existing conditions. By using this concept, we have obtained the optimal temperature profiles.

Integration of the Riccati Equation:

The integration of this equation is carried out using a fourth-order Runge-Kutta method. From the computational point of view, this is a very important equation since almost all the variables involved in Eq.(37) are obtained by solving this equation. The integration poses both a computational as well as a memory storage problem because of the following reasons: As in the integration of the state and adjoint variables, if a step size of 2×10^{-4} hr is used, there is a problem of memory storage since $\underline{P}$ is a (4x4) matrix and it is difficult to store the 16 elements at 5000 points of time. If the step size is increased to 1×10^{-3} hr or more, then there is a problem of numerical instability and the integration using fourth-order Runge Kutta method could not be carried out. In order to overcome both of the above problems, we used a Δt of 2×10^{-4} hr in the integrations, but stored the values of $\underline{P}$ at every time interval of 1×10^{-2} hr i.e. at 101 points of time. The values of $\underline{R}$, $(\frac{\partial H}{\partial T})$, $(\frac{\partial^2 H}{\partial T^2})$, $\frac{\partial f_i}{\partial T}$ are also stored at 101 points of time. The stored values of state and adjoint variables at 2001 points of time are used in the above integrations, and the intermediate values of these are obtained by linear interpolation. In this way we would get stable solutions. One important point to be mentioned is that, in the integration of this eqn., wherever $\underline{P} \to \infty$ for any t', where $0 \le t' < t_f$, we stopped the integration at this point and assumed a value of zero for all the elements

of $\underline{P}$ at lesser times i.e. for $0 \leq t \leq t'$.

Integration of the q Equations:

The integration of these equations are carried out using a fourth-order Runge-Kutta method. Unlike the Riccate equation, this equation poses only a memory storage problem since $\underline{q}$ is a (4x1) vector function of time. In order to overcome this problem, in the integrations, Δt of 1×10^{-3} hr is used and the 4 elements of q are stored at every time interval of 1×10^{-2} hr. We could get stable solutions using the above step size and also the main advantage of using this step size is that the values of state and adjoint variables, available in the memory at every time interval of 1×10^{-3} hr can be directly used in the integrations without any linear interpolation.

Integration of the Perturbation Equations of State Variables:

These equations result from the linearization (Taylor-series expansions to first order) of the non-linear state variable equations and are given in Eq.(34) with the initial conditions given in Eq.(36). The integration of these equations is difficult since both $\delta\underline{X}$ and δT are unknown quantities. In order to find $\delta\underline{X}$, we should know δT before-hand but since our main problem is to correct the temperature profile using Eq.(37), δT is also unknown and hence calculation of $\delta\underline{X}$ is not possible. Normally, to find $\delta\underline{X}$, we have to solve a two-point boundary value problem formed by the equations

$$\delta \overset{\circ}{\underline{X}} = \underline{A}(t) \ \delta \underline{X} - \underline{B}(t) \ \delta \underline{\lambda} \tag{39}$$

$$\delta \overset{\circ}{\underline{\lambda}} = -\underline{C}(t) \ \delta \underline{X} - \underline{A}^T(t) \ \delta \underline{\lambda} \tag{40}$$

where:

$$\underline{A}(t) = (\frac{\partial \underline{f}}{\partial \underline{X}}) - (\frac{\partial \underline{f}}{\partial \underline{T}}) \ (\frac{\partial^2 H}{\partial T^2})^{-1}(\frac{\partial^2 H}{\partial T \ \partial \underline{X}}) \tag{a}$$

$$\underline{B}(t) = (\frac{\partial \underline{f}}{\partial T}) \ (\frac{\partial^2 H}{\partial T^2})^{-1} \ (\frac{\partial \underline{f}}{\partial T})^T \tag{b}$$

$$\underline{C}(t) = \frac{\partial^2 H}{\partial \underline{X}^2} - (\frac{\partial^2 H}{\partial \underline{X} \ \partial T}) \ (\frac{\partial^2 H}{\partial T^2})^{-1} \ (\frac{\partial^2 H}{\partial T \ \partial \underline{X}}) \tag{c} \tag{41}$$

The solution of the above two-point boundary value problem is time-consuming as we have to first eliminate either $\delta \underline{X}$ or $\delta \underline{\lambda}$ from one of the above equations using some simplified assumption. As suggested by Bryson and Ho, we used the backward sweep method in which the solution for $\delta \underline{\lambda}$ is assumed to be of the form

$$\delta \underline{\lambda} = \underline{P} \ \delta \underline{X} \tag{42}$$

where $\underline{P}$ is obtained by solving the Riccati equation. Since the value of $\delta \underline{X}$ at t=0 is known, the value of $\delta \underline{\lambda}$ at t=0 can be found out and then the equations (39) and (40) can be integrated forward as an initial-value problem. Since this is a very tedious process, another way of doing this is to simply substitute Eq.(42) into Eq.(39) and then integrate Eq.(39) forward as an initial-value problem. But we found that this procedure worked only when there are perturbations in the

initial conditions of the state variables. Hence the method for solving $\delta \underline{X}$ we finally adopted is this, in which the expression for $\overline{\delta T}$ $(= T^{(j+1)} - T^{(j)})$ given by Eq.(37) is simply substituted in Eq.(39) and then integrated this equation forward as an initial value problem. The values of $\delta \underline{X}$ so obtained are substituted back into Eq.(37) and the actual correction in the temperature profile is obtained. This kind of method worked for our problem since the values of $\delta \underline{X}$ obtained were of the order of 10^{-7} to 10^{-2}, which are consistent for assuring neighbouring extremal paths.

A listing of the computer program for the second variation method is provided in Appendix.

TABLE 6

DETAILED EQUATIONS FOR THE TERMS $\dfrac{\partial^2 J}{\partial \underline{X}^2}$, $\dfrac{\partial^2 f_i}{\partial \underline{X}^2}$

$$
\frac{\partial^2 J}{\partial \underline{X}^2} =
\begin{bmatrix}
\dfrac{\partial^2 J}{\partial e_g^2} & \dfrac{\partial^2 J}{\partial e_g \partial e_c} & \dfrac{\partial^2 J}{\partial e_g \partial d} & \dfrac{\partial^2 J}{\partial e_g \partial e_v} \\[2em]
\dfrac{\partial^2 J}{\partial e_c \partial e_g} & \dfrac{\partial^2 J}{\partial e_c^2} & \dfrac{\partial^2 J}{\partial e_c \partial d} & \dfrac{\partial^2 J}{\partial e_c \partial e_v} \\[2em]
\dfrac{\partial^2 J}{\partial d \partial e_g} & \dfrac{\partial^2 J}{\partial d \partial e_c} & \dfrac{\partial^2 J}{\partial d^2} & \dfrac{\partial^2 J}{\partial d \partial e_v} \\[2em]
\dfrac{\partial^2 J}{\partial e_v \partial e_g} & \dfrac{\partial^2 J}{\partial e_v \partial e_c} & \dfrac{\partial^2 J}{\partial e_v \partial d} & \dfrac{\partial^2 J}{\partial e_v^2}
\end{bmatrix}
$$

$$
\frac{\partial^2 J}{\partial e_g^2} = \frac{2\,\alpha_2\,e_{g\bullet}}{\mu_{nd}^2}\left(\frac{3\,e_{go}}{eg^4} - \frac{2\mu_{nd}}{e_g^3}\right)
$$

$$
\frac{\partial^2 J}{\partial e_c^2} = \frac{\partial^2 J}{\partial e_v^2} = 2\,\alpha_3
$$

The remaining elements are zero

$\dfrac{\partial^2 f_1}{\partial \underline{X}^2}$ - This is a (4x4) matrix of the same form as $\left(\dfrac{\partial^2 J}{\partial \underline{X}^2}\right)$

with J replaced everywhere by f_1. The detailed

equations for the elements involved in this matrix

are:

$$\frac{\partial^2 f_1}{\partial e_g{}^2} = - \frac{4\,k_3}{V} - \frac{4\,k_6}{V} - \left(\frac{8\,k_3}{V\,K_3} + \frac{4\,k_5}{V} \right) \frac{\partial g}{\partial e_g}$$

$$-\left(\frac{2\,k_7}{V\,K_4} + \frac{2\,k_8}{V\,K_5} \right) \frac{\partial w}{\partial e_g}$$

$$\frac{\partial^2 f_1}{\partial e_c \partial e_g} = - \left(\frac{4\,k_3}{V\,K_3} + \frac{2\,k_5}{V} \right) \frac{\partial g}{\partial e_c} - \left(\frac{4\,k_3}{V\,K_3} - \frac{2\,k_7}{V} \right) \frac{\partial g}{\partial e_g}$$

$$- \left(\frac{k_7}{V\,K_4} + \frac{k_8}{V\,K_5} \right) \frac{\partial w}{\partial e_c} - \frac{k_8}{V} \left(1 + \frac{1}{K_5} \frac{\partial w}{\partial e_g} \right)$$

$$\frac{\partial^2 f_1}{\partial d\, \partial e_g} = 0.0$$

$$\frac{\partial^2 f_1}{\partial e_v\, \partial e_c} = \left(- \frac{4\,k_3}{V\,K_3} + \frac{2\,k_7}{V} \right) \frac{\partial g}{\partial e_v} - \frac{4\,k_3}{V\,K_3} \frac{\partial g}{\partial e_c}$$

$$- \frac{k_8}{V K_5} \left(\frac{\partial w}{\partial e_v} + \frac{\partial w}{\partial e_c} \right)$$

$$\frac{\partial^2 f_1}{\partial e_g \partial d} = \frac{\partial^2 f_1}{\partial e_c\, \partial d} = \frac{\partial^2 f_1}{\partial d^2} = \frac{\partial^2 f_1}{\partial e_v\, \partial d} = 0.0$$

$$\frac{\partial^2 f_1}{\partial e_v{}^2} = - \frac{8\,k_3}{V K_3} \frac{\partial g}{\partial e_v} - \frac{2\,k_8}{V K_5} \frac{\partial w}{\partial e_v}$$

The remaining elements are obtained by noting that

$$\frac{\partial^2 f_1}{\partial X_i\, \partial X_j} = \frac{\partial^2 f_1}{\partial X_j\, \partial X_i} \quad \text{where} \quad i,j = 1,2,3,4$$

$\dfrac{\partial^2 f_2}{\partial \underline{X}^2}$ - This is a (4x4) matrix of the same form as $\left(\dfrac{\partial^2 J}{\partial \underline{X}^2} \right)$ with

$\underline{J}$ replaced everywhere by f_2. The detailed equations for the elements involved in this matrix are:

$$\frac{\partial^2 f_2}{\partial e_g^2} = \frac{4\,k_5}{V}\ \frac{\partial g}{\partial e_g} + \frac{2\,k_6}{V} + \frac{2\,k_7}{V\,K_4}\ \frac{\partial w}{\partial e_g} - \frac{2\,k_8}{V K_5}\ \frac{\partial w}{\partial e_g}$$

$$\frac{\partial^2 f_2}{\partial e_c\,\partial e_g} = \frac{2\,k_5}{V}\ \frac{\partial g}{\partial e_c} - \frac{2\,k_7}{V}\ \frac{\partial g}{\partial e_g} + \left(\frac{k_7}{V K_4} - \frac{k_8}{V K_5}\right)\ \frac{\partial w}{\partial e_c}$$

$$- \frac{k_8}{V}\left(1 + \frac{1}{K_5}\ \frac{\partial w}{\partial e_g}\right)$$

$$\frac{\partial^2 f_2}{\partial d\,\partial e_g} = 0.0$$

$$\frac{\partial^2 f_2}{\partial e_v\,\partial e_g} = \frac{2\,k_5}{V}\ \frac{\partial g}{\partial e_v} + \left(\frac{k_7}{V K_4} - \frac{k_8}{V K_5}\right)\ \frac{\partial w}{\partial e_v} - \frac{k_8}{V K_5}\ \frac{\partial w}{\partial e_g}$$

$$\frac{\partial^2 f_2}{\partial e_c^2} = -\frac{4\,k_7}{V}\ \frac{\partial g}{\partial e_c} - \frac{2\,k_8}{V K_5}\ \frac{\partial w}{\partial e_c}$$

$$\frac{\partial^2 f_2}{\partial d\,\partial e_c} = 0.0$$

$$\frac{\partial^2 f_2}{\partial e_v\,\partial e_c} = -\frac{2\,k_7}{V}\ \frac{\partial g}{\partial e_v} - \frac{k_8}{V K_5}\left(\frac{\partial w}{\partial e_v} + \frac{\partial w}{\partial e_c}\right)$$

$$\frac{\partial^2 f_2}{\partial e_g\,\partial d} = \frac{\partial^2 f_2}{\partial e_c\,\partial d} = \frac{\partial^2 f_2}{\partial d^2} = \frac{\partial^2 f_2}{\partial e_v\,\partial d} = 0.0$$

$$\frac{\partial^2 f_2}{\partial e_v^2} = -\frac{2\,k_8}{V K_5}\ \frac{\partial w}{\partial e_v}$$

The remaining elements are obtained by noting that

$$\frac{\partial^2 f_2}{\partial X_i \, \partial X_j} = \frac{\partial^2 f_2}{\partial X_j \, \partial X_i} \quad \text{where } i,j = 1,2,3,4$$

$\dfrac{\partial^2 f_3}{\partial \underline{X}^2}$ — This is a (4x4) matrix of the same form as $\left(\dfrac{\partial^2 \underline{J}}{\partial \underline{X}^2}\right)$ with

$\underline{J}$ replaced everywhere by f_3. The detailed equations for the elements involved in this matrix are:

$$\frac{\partial^2 f_3}{\partial e_g^2} = \frac{4 \, k_5}{V} \, \frac{\partial g}{\partial e_g} + \frac{2 \, k_6}{V}$$

$$\frac{\partial^2 f_3}{\partial e_c \, \partial e_g} = \frac{\partial^2 f_3}{\partial e_g \, \partial e_c} = \frac{2 \, k_5}{V} \, \frac{\partial g}{\partial e_c}$$

$$\frac{\partial^2 f_3}{\partial d \, \partial e_g} = 0.0$$

$$\frac{\partial^2 f_3}{\partial e_v \, \partial e_g} = \frac{\partial^2 f_3}{\partial e_g \, \partial e_v} = \frac{2 \, k_5}{V} \, \frac{\partial g}{\partial e_v}$$

The remaining elements are zero.

$\dfrac{\partial^2 f_4}{\partial \underline{X}^2}$ — This is a (4x4) matrix of the same form as $\left(\dfrac{\partial^2 \underline{J}}{\partial \underline{X}^2}\right)$ with

$\underline{J}$ replaced everywhere by f_4. The detailed equations for the elements involved in this matrix are:

$$\frac{\partial^2 f_4}{\partial e_v \, \partial e_g} = \frac{\partial^2 f_4}{\partial e_g \, \partial e_v} = -\frac{k_3}{V}$$

The remaining elements are zero.
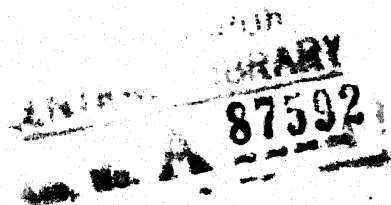
TABLE 7

DETAILED EQUATIONS FOR THE TERMS $\dfrac{\partial^2 J}{\partial T\,\partial \underline{X}}$, $\dfrac{\partial^2 f_1}{\partial T\,\partial \underline{X}}$

$$\frac{\partial^2 J}{\partial T\,\partial \underline{X}} = \left[\frac{\partial^2 J}{\partial T\,\partial e_g} \; , \; \frac{\partial^2 J}{\partial T\,\partial c_c} , \; \frac{\partial^2 J}{\partial T\,\partial d} , \; \frac{\partial^2 J}{\partial T\,\partial e_v}\right]^T$$

All the elements in the above matrix are zero.

$\dfrac{\partial^2 f_1}{\partial T\,\partial \underline{X}}$ — This is a (4x1) vector of the same form as $\left(\dfrac{\partial^2 J}{\partial T\,\partial \underline{X}}\right)$ with $\underline{J}$ replaced everywhere by f_1. The detailed equations for the elements involved in this vector are:

$$\frac{\partial^2 f_1}{\partial T\,\partial e_g} = (-4\,e_g - e_v)\,\frac{d(k_3/V)}{dT} - \frac{4}{K_3}\left[\left(\frac{k_3}{V}\right)\frac{\partial g}{\partial T} + g\,\frac{d(k_3/V)}{dT}\right]$$

$$+ \frac{4\,(e_{go}-e_g-e_c-e_v)}{K_3}\left[\left(\frac{k_3}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_g} + \frac{\partial g}{\partial e_g}\,\frac{d(k_3/V)}{dT}\right]$$

$$- \frac{d\,(k_4)}{dT} - 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_g} + \frac{\partial g}{\partial e_g}\,\frac{d(k_5/V)}{dT}\right]$$

$$- 2\left[\left(\frac{k_5}{V}\right)\frac{\partial g}{\partial T} + g\,\frac{d(k_5/V)}{dT}\right] - 4\,e_g\,\frac{d(k_5/V)}{dT}$$

$$+ 2\,e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_g} + \frac{\partial g}{\partial e_g}\,\frac{d(k_7/V)}{dT}\right] - \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 w}{\partial T\,\partial e_g}\right.$$

$$\left.+ \frac{\partial w}{\partial e_g}\,\frac{d(k_7/V)}{dT}\right] - \frac{1}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial w}{\partial T} + w\,\frac{d(k_7/V)}{dT}\right]$$

$$- e_c\,\frac{d(k_8/V)}{dT} - \frac{1}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial w}{\partial T} + w\,\frac{d(k_8/V)}{dT}\right]$$

$$\frac{\partial^2 f_1}{\partial T\,\partial e_c} = -\frac{4}{K_3}\left[\left(\frac{k_3}{V}\right)\frac{\partial g}{\partial T} + g\,\frac{d(k_3/V)}{dT}\right] + \frac{4\,(e_{go}-e_g-e_c-e_v)}{K_3}$$

$$[(\frac{k_3}{V}) \frac{\partial^2 g}{\partial T \partial e_c} + \frac{\partial g}{\partial e_c} \frac{d(k_3/V)}{dT}] - 2e_g [(\frac{k_5}{V}) \frac{\partial^2 g}{\partial T \partial e_c}$$

$$+ \frac{\partial g}{\partial e_c} \frac{d(k_5/V)}{dT}] + 2e_c [(\frac{k_7}{V}) \frac{\partial^2 g}{\partial T \partial e_c} + \frac{\partial g}{\partial e_c} \frac{d(k_7/V)}{dT}]$$

$$+ 2 [(\frac{k_7}{V}) \frac{\partial g}{\partial T} + g \frac{d(k_7/V)}{dT}] - \frac{e_g}{K_4}[(\frac{k_7}{V}) \frac{\partial^2 w}{\partial T \partial e_c} +$$

$$\frac{\partial w}{\partial e_c} \frac{d(k_7/V)}{dT}] - e_g \frac{d(k_8/V)}{dT} - \frac{1}{K_5}[(\frac{k_8}{V}) \frac{\partial w}{\partial T} + w \frac{d(k_8/V)}{dT}]$$

$$+ \frac{(e_{go}-e_g-e_c-e_v)}{K_5} [(\frac{k_8}{V}) \frac{\partial^2 w}{\partial T \partial e_c} + \frac{\partial w}{\partial e_c} \frac{d(k_8/V)}{dT}]$$

$$\frac{\partial^2 f_1}{\partial T \partial d} = 0.0$$

$$\frac{\partial^2 f_1}{\partial T \partial e_v} = - \frac{4}{K_3}[(\frac{k_3}{V}) \frac{\partial g}{\partial T} + g \frac{d(k_3/V)}{dT}]$$

$$+ \frac{4(e_{go}-e_g-e_c-e_v)}{K_3} [(\frac{k_3}{V}) \frac{\partial^2 g}{\partial T \partial e_v} + (\frac{\partial g}{\partial c_v}) \frac{d(k_3/V)}{dT}]$$

$$- 2e_g [(\frac{k_5}{V}) \frac{\partial^2 g}{\partial T \partial e_v} + (\frac{\partial g}{\partial e_v}) \frac{d(k_5/V)}{dT}]$$

$$+ 2e_c [(\frac{k_7}{V}) \frac{\partial^2 g}{\partial T \partial e_v} + (\frac{\partial g}{\partial e_v}) \frac{d(k_7/V)}{dT}]$$

$$- \frac{e_g}{K_4} [(\frac{k_7}{V}) \frac{\partial^2 w}{\partial T \partial e_v} + \frac{\partial w}{\partial e_v} \frac{d(k_7/V)}{dT}]$$

$$- \frac{1}{K_5} [(\frac{k_8}{V}) \frac{\partial w}{\partial T} + w \frac{d(k_8/V)}{dT}] + \frac{(e_{go}-e_g-e_c-e_v)}{K_5}$$

$$[(\frac{k_8}{V}) \frac{\partial^2 w}{\partial T \partial e_v} + (\frac{\partial w}{\partial e_v}) \frac{d(k_8/V)}{dT}] - e_g \frac{d(k_3/V)}{dT}$$

$\dfrac{\partial^2 f_2}{\partial T\,\partial \underline{X}}$ - This is a (4x1) vector of the same form as $(\dfrac{\partial^2 J}{\partial T\,\partial \underline{X}})$ with $\underline{J}$ replaced everywhere by f_2. The detailed equations for the elements involved in this vector are:

$$\frac{\partial^2 f_2}{\partial T\,\partial e_g} = \frac{d(k_4)}{dT} + 2\,e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_g} + \frac{\partial g}{\partial e_g}\frac{d(k_5/V)}{dT}\right]$$

$$+2\left[\left(\frac{k_5}{V}\right)\frac{\partial g}{\partial T} + g\,\frac{d(k_5/V)}{dT}\right] + 2\,e_g\,\frac{d(k_6/V)}{dT}$$

$$- 2\,e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_g} + \frac{\partial g}{\partial e_g}\frac{d(k_7/V)}{dT}\right]$$

$$+ \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 w}{\partial T\,\partial e_g} + \frac{\partial w}{\partial e_g}\frac{d(k_7/V)}{dT}\right]$$

$$+ \frac{1}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial w}{\partial T} + w\,\frac{d(k_7/V)}{dT}\right] - e_c\,\frac{d(k_8/V)}{dT}$$

$$- \frac{1}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial w}{\partial T} + w\,\frac{d(k_8/V)}{dT}\right] + \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial^2 w}{\partial T\,\partial e_g}\right.$$

$$\left.+ \frac{\partial w}{\partial e_g}\frac{d(k_8/V)}{dT}\right] - \frac{1}{2}\frac{d(k_9)}{dT}$$

$$\frac{\partial^2 f_2}{\partial T\,\partial e_c} = 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_c} + \frac{\partial g}{\partial e_c}\frac{d(k_5/V)}{dT}\right]$$

$$-2e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 g}{\partial T\,\partial e_c} + \frac{\partial g}{\partial e_c}\frac{d(k_7/V)}{dT}\right]$$

$$-2\left[\left(\frac{k_7}{V}\right)\frac{\partial g}{\partial T} + g\,\frac{d(k_7/V)}{dT}\right] + \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 w}{\partial T\,\partial e_c}\right.$$

$$\left.+ \frac{\partial w}{\partial e_c}\frac{d(k_7/V)}{dT}\right] - e_g\,\frac{d(k_8/V)}{dT} - \frac{1}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial w}{\partial T}\right.$$

$$\left.+ w\,\frac{d(k_8/V)}{dT}\right] + \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial^2 w}{\partial T\,\partial e_c}\right.$$

$$+ \frac{\partial w}{\partial e_c} \frac{d(k_8/V)}{dT}] - \frac{1}{2} \frac{d(k_9)}{dT}$$

$$\frac{\partial^2 f_2}{\partial T \partial d} = 0.0$$

$$\frac{\partial^2 f_2}{\partial T \partial e_v} = 2e_g [(\frac{k_5}{V}) \frac{\partial^2 g}{\partial T \partial e_v} + \frac{\partial g}{\partial e_v} \frac{d(k_5/V)}{dT}]$$

$$-2 e_c [(\frac{k_7}{V}) \frac{\partial^2 g}{\partial T \partial e_v} + \frac{\partial g}{\partial e_v} \frac{d(k_7/V)}{dT}]$$

$$+ \frac{e_g}{K_4} [(\frac{k_7}{V}) \frac{\partial^2 w}{\partial T \partial e_v} + \frac{\partial w}{\partial e_v} \frac{d(k_7/V)}{dT}]$$

$$- \frac{1}{K_5} [(\frac{k_8}{V}) \frac{\partial w}{\partial T} + w \frac{d(k_8/V)}{dT}] +$$

$$(\frac{e_{g0} - e_g - e_c - e_v}{K_5}) [(\frac{k_8}{V}) \frac{\partial^2 w}{\partial T \partial e_v} + \frac{\partial w}{\partial e_v} \frac{d(k_8/V)}{dT}] - \frac{1}{2} \frac{d(k_9)}{dT}$$

$$\frac{\partial^2 f_3}{\partial T \partial \underline{X}}$$ — This is a (4x1) vector of the same form as $(\frac{\partial^2 J}{\partial T \partial \underline{X}})$ with J replaced everywhere by f_3. The detailed equations for the elements involved in this vector are:

$$\frac{\partial^2 f_3}{\partial T \partial e_g} = 2_{eg} [(\frac{k_5}{V}) \frac{\partial^2 g}{\partial T \partial e_g} + \frac{\partial g}{\partial e_g} \frac{d(k_5/V)}{dT}]$$

$$+ 2 [(\frac{k_5}{V}) \frac{\partial g}{\partial T} + ng \frac{d(k_5/V)}{dT}] + 2e_g \frac{d(k_6/V)}{dT}$$

$$\frac{\partial^2 f_3}{\partial T \partial e_c} = 2 e_g [(\frac{k_5}{V}) \frac{\partial^2 g}{\partial T \partial e_c} + \frac{\partial g}{\partial e_c} \frac{d(k_5/V)}{dT}]$$

$$\frac{\partial^2 f_3}{\partial T \partial d} = 0.0$$

$$\frac{\partial^2 f_3}{\partial T \partial e_v} = 2 e_g \left[\left(\frac{k_5}{V} \right) \frac{\partial^2 g}{\partial T \partial e_v} + \frac{\partial g}{\partial e_v} \frac{d(k_5/V)}{dT} \right]$$

$\dfrac{\partial^2 f_4}{\partial T \partial \underline{X}}$ - This is a (4x1) vector of the same form as $\left(\dfrac{\partial^2 J}{\partial T \partial \underline{X}} \right)$ with $\underline{J}$ replaced everywhere by f_4. The detailed equations for the elements involved in this vector are:

$$\frac{\partial^2 f_4}{\partial T \partial e_g} = -\frac{1}{2} \frac{d(k_9)}{dT} - e_v \frac{d(k_3/V)}{dT}$$

$$\frac{\partial^2 f_4}{\partial T \partial e_c} = -\frac{1}{2} \frac{d(k_9)}{dT}$$

$$\frac{\partial^2 f_4}{\partial T \partial d} = 0.0$$

$$\frac{\partial^2 f_4}{\partial T \partial e_v} = -\frac{1}{2} \frac{d(k_9)}{dT} - e_g \frac{d(k_3/V)}{dT}$$

$$\frac{\partial^2 g}{\partial T \partial e_g} = \frac{\partial^2 g}{\partial T \partial e_c} = \frac{\partial^2 g}{\partial T \partial d} = \frac{\partial^2 g}{\partial T \partial e_v} = -\frac{P_T}{2(P_W^\circ - P_T)^2} \frac{d P_G^\circ}{dT}$$

$$\frac{\partial^2 w}{\partial T \partial e_g} = \frac{\partial^2 w}{\partial T \partial e_c} = \frac{\partial^2 w}{\partial T \partial d} = \frac{\partial^2 w}{\partial T \partial e_v} = -\frac{P_T}{2(P_W^\circ - P_T)^2} \frac{dP_W^\bullet}{dT}$$

TABLE 8

DETAILED EQUATIONS FOR THE TERMS $\dfrac{\partial^2 f_i}{\partial T^2}$

$$\frac{\partial^2 f_1}{\partial T^2} = -2e_g^2 \frac{d^2(k_3/V)}{dT^2} + \frac{4(e_{go}-e_g-e_c-e_v)}{K_3}\left[\left(\frac{k_3}{V}\right)\frac{\partial^2 g}{\partial T^2}\right.$$

$$+ g\frac{d^2(k_3/V)}{dT^2} + 2\frac{d(k_3/V)}{dT}\frac{\partial g}{\partial T}\right] - e_g\frac{d^2(k_4)}{dT^2}$$

$$- 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial^2 g}{\partial T^2} + g\frac{d^2(k_5/V)}{dT^2} + 2\frac{d(k_5/V)}{dT}\frac{\partial g}{\partial T}\right]$$

$$- 2e_g^2\frac{d^2(k_6/V)}{dT^2} + 2e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 g}{\partial T^2} + g\frac{d^2(k_7/V)}{dT^2}\right.$$

$$+ 2\frac{d(k_7/V)}{dT}\frac{\partial g}{\partial T}\right] - \frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 w}{\partial T^2} + w\frac{d^2(k_7/V)}{dT^2}\right.$$

$$+ 2\frac{d(k_7/V)}{dT}\frac{\partial w}{\partial T}\right] - e_c e_g\frac{d^2(k_8/V)}{dT^2} + \frac{(e_{go}-e_g-e_c-e_v)}{K_5}$$

$$\left[\left(\frac{k_8}{V}\right)\frac{\partial^2 w}{\partial T^2} + w\frac{d^2(k_8/V)}{dT^2} + 2\frac{d(k_8/V)}{dT}\frac{\partial w}{\partial T}\right]$$

$$- e_v e_g\frac{d^2(k_3/V)}{dT^2}$$

$$\frac{\partial^2 f_2}{\partial T^2} = e_g\frac{d^2(k_4)}{dT^2} + 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial g}{\partial T} + \frac{d(k_5/V)}{dT}\frac{\partial g}{\partial T} + \right.$$

$$g\frac{d(k_5/V)}{dT} + g\frac{d^2(k_5/V)}{dT^2}\right] + e_g^2\frac{d^2(k_6/V)}{dT^2}$$

$$- 2e_c\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 g}{\partial T^2} + g\frac{d^2(k_7/V)}{dT^2} + 2\frac{d(k_7/V)}{dT}\frac{\partial g}{\partial T}\right]$$

$$+\frac{e_g}{K_4}\left[\left(\frac{k_7}{V}\right)\frac{\partial^2 w}{\partial T^2} + w\frac{d^2(k_7/V)}{dT^2} + 2\frac{d(k_7/V)}{dT}\frac{\partial w}{\partial T}\right]$$

$$-e_c\,e_g\frac{d^2(k_8/V)}{dT^2} + \frac{(e_{go}-e_g-e_c-e_v)}{K_5}\left[\left(\frac{k_8}{V}\right)\frac{\partial^2 w}{\partial T^2} + \right.$$

$$w\frac{d^2(k_8/V)}{dT^2} + 2\frac{d(k_8/V)}{dT}\frac{\partial w}{\partial T}\Big] + \frac{(e_{go}-e_g-e_c-e_v)}{2}\frac{d^2(k_9)}{dT^2}$$

$$\frac{\partial^2 f_3}{\partial T^2} = 2e_g\left[\left(\frac{k_5}{V}\right)\frac{\partial^2 g}{\partial T^2} + g\frac{d^2(k_5/V)}{dT^2} + 2\frac{d(k_5/V)}{dT}\frac{\partial g}{\partial T}\right]$$

$$+ e_g^2\frac{d^2(k_6/V)}{dT^2}$$

$$\frac{\partial^2 f_4}{\partial T^2} = \frac{(e_{go}-e_g-e_c-e_v)}{2}\frac{d^2(k_9)}{dT^2} - e_v e_g\frac{d^2(k_3/V)}{dT^2}$$

$$\frac{\partial^2 g}{\partial T^2} = \frac{-g\dfrac{d^2 P_G^\circ}{dT^2} - w\dfrac{d^2 P_W^\circ}{dT^2} - 2\dfrac{dP_G^\circ}{dT}\dfrac{\partial g}{\partial T}}{(P_G^\circ - P_T)}$$

$$\frac{\partial^2 w}{\partial T^2} = \frac{-g\dfrac{d^2 P_G^\circ}{dT^2} - w\dfrac{d^2 P_W^\circ}{dT^2} - 2\dfrac{dP_W^\circ}{dT}\dfrac{\partial w}{\partial T}}{(P_W^\circ - P_T)}$$

$$\frac{d^2 P_G^\circ}{dT^2} = P_G^\circ\left[\frac{-7458}{T^3} + \frac{4.042}{T}\right] + \left[\frac{3729}{T^2} - \frac{4.042}{T}\right]\frac{dP_G^\circ}{dT}$$

$$\frac{d^2 P_W^\circ}{dT^2} = P_W^\circ\left[-\frac{3515.706}{(T-33.274)^3}\right] + \left[\frac{1757.853}{(T-33.274)^2}\right]\frac{dP_W^\bullet}{dT}$$

$$\frac{d^2(k_i/V)}{dT^2} = \frac{1}{VT^2}\frac{dk_i}{dT}\left\{\frac{E_i}{R} - \left(2T + \frac{T^2}{V}\frac{dV}{dT}\right)\right\}$$

$$- \frac{1}{V^2} \left(\frac{dV}{dT}\right) \frac{dk_i}{dT} - \frac{2k_i}{V} \left(\frac{dV}{dT}\right)$$

$$- \frac{k_i}{V^2} \frac{d^2 V}{dT^2} \quad , \quad i = 3, 5, 6, 7, 8$$

$$\frac{d^2(k_i)}{dT^2} = \frac{1}{T} \frac{dk_i}{dT} \left[\frac{E_i}{RT} - 2\right] \quad , \quad i = 4, 9$$

$$\frac{d k_i}{dT} = \frac{k_i E_i}{RT^2} \quad , \quad i = 3, \ldots, 9$$

$$\frac{d^2 V}{dT^2} = V_G \frac{d^2 X_G}{dT^2} + 2 \left(\frac{d V_G}{dT}\right) \left(\frac{dX_G}{dT}\right) + V_w \frac{d^2 X_w}{dT^2}$$

$$+ 2\left(\frac{dV_w}{dT}\right) \left(\frac{dX_w}{dT}\right)$$

$$\frac{d^2 X_G}{dT^2} = \left[\frac{\frac{(e_g+e_c+d+e_v)}{2} + w + a}{\left[\frac{(e_g+e_c+d+e_v)}{2} + g + w + a\right]^2}\right] \frac{\partial^2 g}{\partial T^2} - \frac{2}{\left[\frac{(e_g+e_c+d+e_v)}{2}+g+w+a\right]} \left(\frac{\partial g}{\partial T}\right)^2$$

$$\frac{d^2 X_w}{dT^2} = \left[\frac{\frac{(e_g+e_c+d+e_v)}{2} + g + a}{\left[\frac{(e_g+e_c+d+e_v)}{2} + g+w+a\right]^2}\right] \frac{\partial^2 w}{\partial T^2} - \frac{2}{\left[\frac{(e_g+e_c+d+e_v)}{2}+g+w+a\right]} \left(\frac{\partial w}{\partial T}\right)^2$$

CHAPTER 4

RESULTS AND DISCUSSION

With the objective function defined in Eq.(11), the optimal temperature profiles in batch reactor are obtained for various reactor pressures using the combined method in which the control vector Iteration method is used for the first few iterations and then switching over to the second variation method till the change in the objective function from iteration to iteration is small. For starting the computations, initial guess temperature should be assumed. In the earlier studies it was found that even though the temperature history is dependent on the initial guess temperature assumed, the value of the objective function does not. Hence, we used an initial guess temperature of 500°K for the control vector iteration method and carried out all the computations. For the second variation method the near optimal temperature profile obtained from the control vector iteration method is used as the initial guess temperature profile.

In the integration of the Eq.(20) or the Riccati equation in the second variation method, with some values of the weighting parameters, we found that in some iteration, $\underline{P} \to \infty$ at t' where $0 \leq t' < t_f$. At this point, as suggested by Bryson and Ho,[46] we stopped the integration of this equation

the objective function dominates and this would give smaller

deviation of μ_n from μ_{nd}. This can happen only if the

conversion of e_g increases. In Fig. 3, almost all the curves

indicate the use of high temperatures initially which must be

brought down towards the end. For low values of α_2, the

curves indicate the use of relatively low temperatures since

the term corresponding to μ_n in the objective function has a

very low value when compared to other terms and the effect of

the other terms is to minimize the formation of side products

and diethylene glycol content in the polymer. This can be

accomplished by bringing down the reactor temperature. In

particular, for a value of $\alpha_2 = 0.5$, the reactor temperature

is seen to approach the lower limit set for the temperature.

In Fig. 4, the increase in the μ_n of the polymer is plotted and

it is seen that in all the curves, μ_n rises very fast and

afterwards remains almost constant at an asymptotic value, which

is attained in a relatively smaller time. It is implied from

this behaviour of μ_n that in the optimization, μ_n reaches μ_{nd}

first and later on the minimization of the other terms in the

objective function is taken care of. It is also found from

Table 9 that the value of μ_n is low for the low values of α_2.

In Fig. 5, for a value of $\alpha_2 = 100$, the optimal temperature

profiles obtained by the combined method and the control vector

iteration method are compared. It is seen that the profiles

are almost the same except in the initial and final stages

TABLE 9

EFFECT OF VARIATION OF α_2 ON THE VARIOUS RESULTS

$\alpha_1 = 1000$

$\alpha_3 = 1000$

$P_T = 200$ mmHg

Amount flashed	μ_n	Conversion, %	e_c	d	e_v	g	w	a
0.000000	1.1395	12.24	0.465×10^{-5}	0.522×10^{-5}	0.297×10^{-10}	0.612×10^{-1}	0.809×10^{-6}	0.243×10^{-6}
0.000000	1.1738	14.81	0.662×10^{-5}	0.785×10^{-5}	0.522×10^{-10}	0.74×10^{-1}	0.16×10^{-5}	0.368×10^{-4}
0.425896	7.1406	86.30	0.493×10^{-5}	0.255×10^{-2}	0.516×10^{-6}	0.542×10^{-2}	0.968×10^{-4}	0.232×10^{-4}
0.457962	11.8351	91.995	0.639×10^{-3}	0.38×10^{-2}	0.692×10^{-6}	0.174×10^{-2}	0.694×10^{-4}	0.338×10^{-4}
0.465662	13.7224	93.20	0.985×10^{-3}	0.39×10^{-2}	0.928×10^{-6}	0.124×10^{-2}	0.897×10^{-4}	0.496×10^{-4}
0.466206	13.5880	93.190	0.107×10^{-2}	0.443×10^{-2}	0.108×10^{-5}	0.125×10^{-2}	0.976×10^{-4}	0.476×10^{-4}
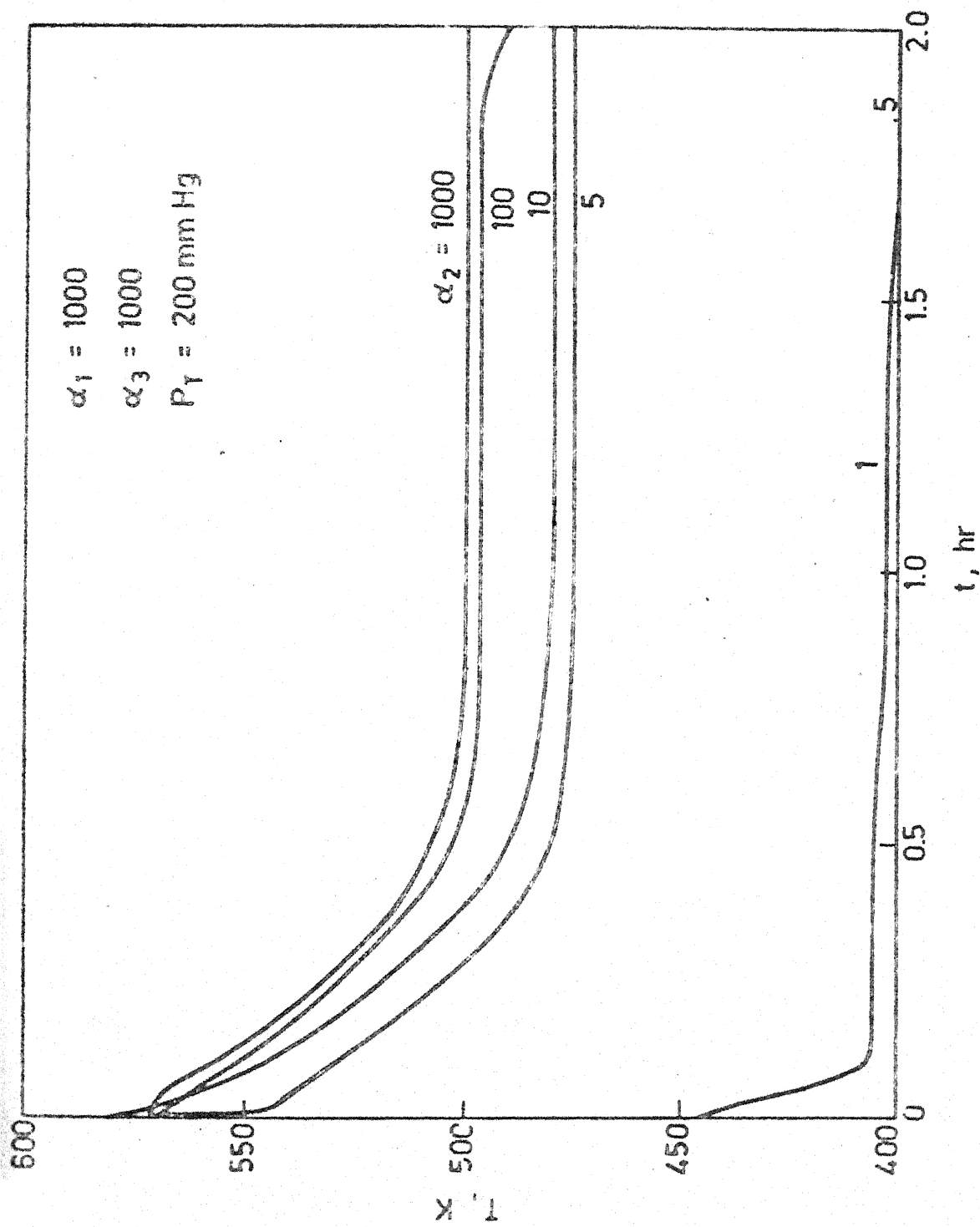
Fig. 3 Effect of variation of α_2 on the optimal temperature profile.
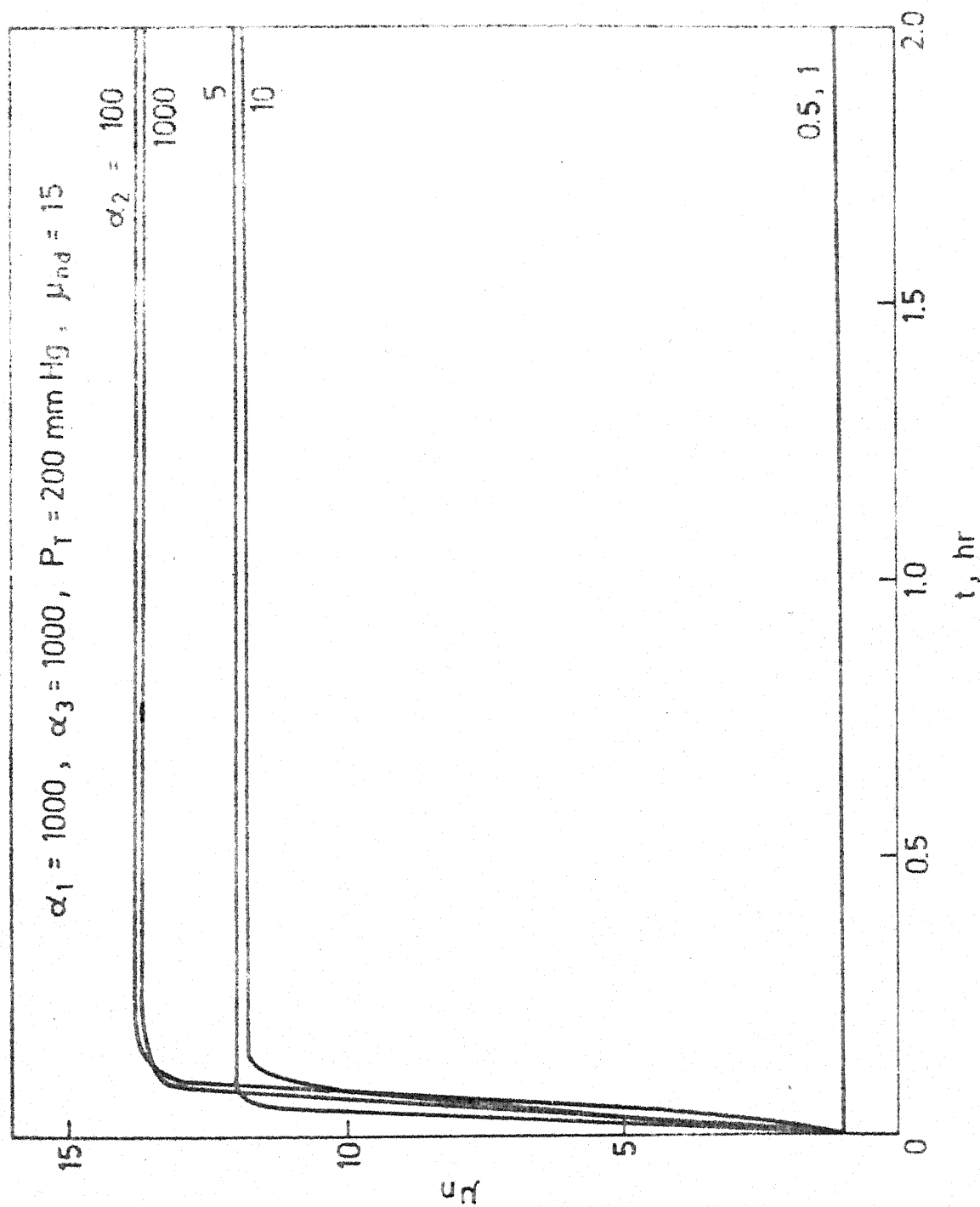
Fig. 4 Variation of μ_n with t for different values of α_2.
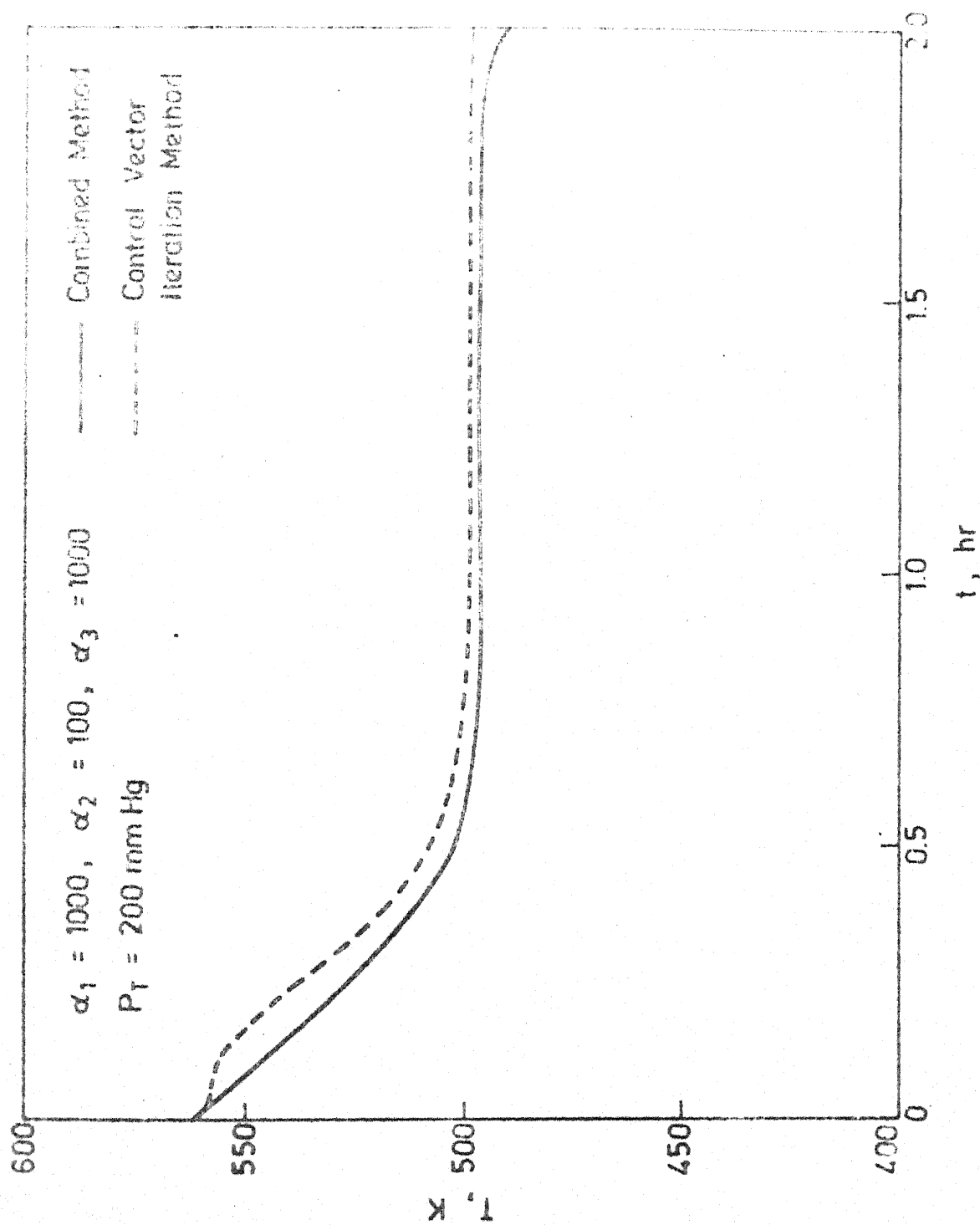
Fig 5 Comparison of the optimal temperature profiles obtained by the combined method and control vector iteration method.
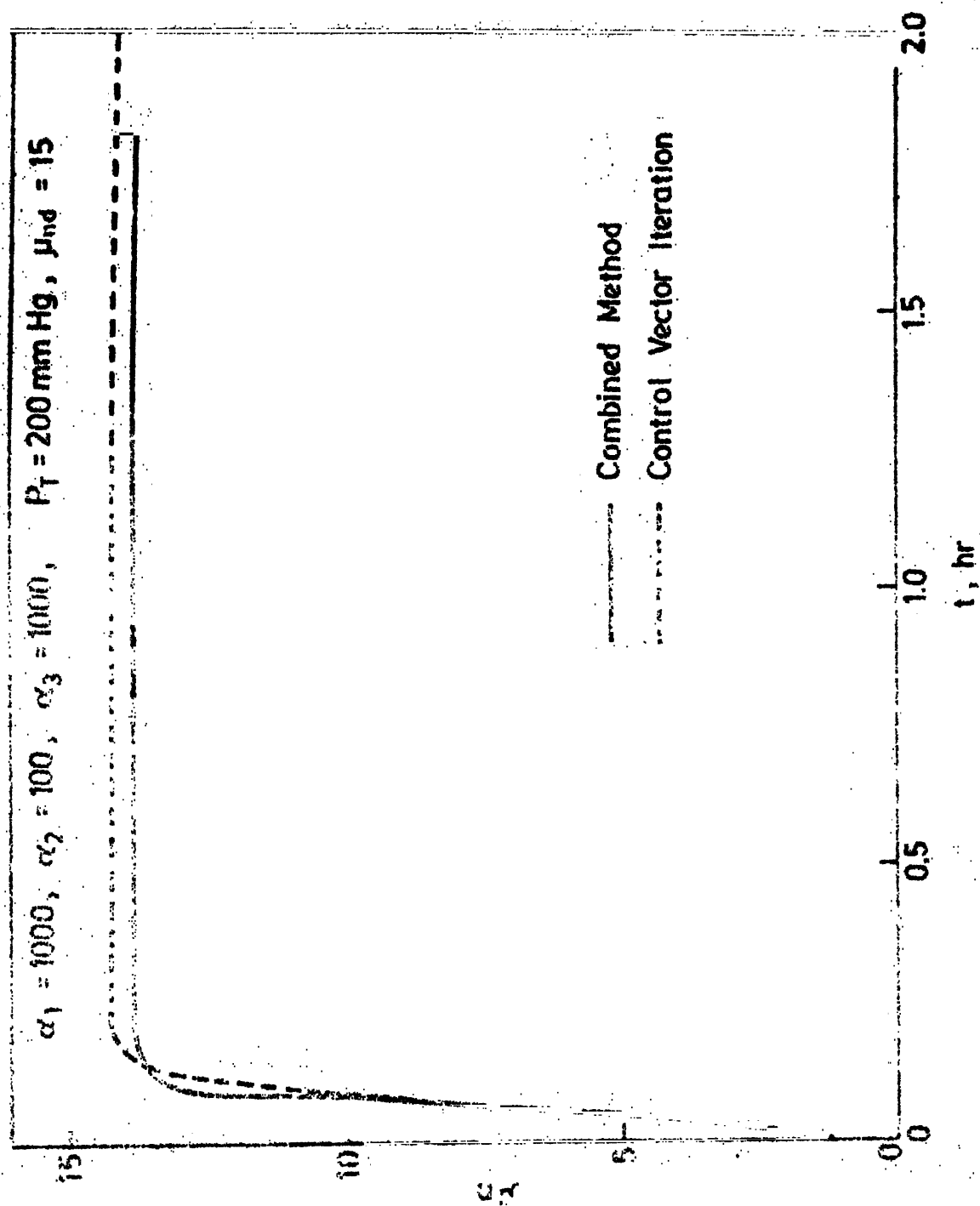
Fig. 6 Comparison of the μ_n obtained by the combined method and the control vector iteration met

wherein the temperatures are slightly more or less. This confirms our contention that the second variation method makes only small changes in the near optimal temperature profile obtained from the control vector iteration method. In Fig. 6, μ_n obtained for a value of α_2=100 by using the ~~true~~ *two* methods are compared and it is seen that the curves almost match in the initial stages,but the asymptotic values are slightly different. This is because of the changes in the optimal temperature profiles in Fig.5.

The weighting parameter α_1 represents the relative importance of the diethylene glycol content in the polymer. By varying α_1, the optimal temperature profiles and the other results obtained are shown in Table 10 and Fig.7. In Table 10, it is seen that as α_1 increases, the conversion of e_g as well as the value of μ_n drops because of the domination of the first term in the objective function when compared to other terms. The result is that the moles of diethylene glycol in the polymer is minimized by having low conversion of E_g groups. Since the conversion is low, the reactor temperature is also low as seen in Fig.7. In this figure, it is also seen that the curves indicate the use of high temperatures initially for low values of α_1, but as α_1 increases, the initial temperatures begin to fall. In the final stages all the curves indicate the use of low temperatures. In particular, for a value of $\alpha_1 = 5000$, the reactor temperature is seen to approach the lower limit

TABLE 10

EFFECT OF VARIATION OF α_1 ON THE VARIOUS RESULTS

$\alpha_2 = 1$ $\alpha_3 = 1000$ $P_T = 200$ mmHg

Amount flashed	μ_n	Conversion,%	e_c	d	e_v	g	w	a	I
0.464306	13.696	93.21	0.105×10^{-2}	0.403×10^{-2}	0.911×10^{-2}	0.124×10^{-2}	0.956×10^{-4}	0.511×10^{-4}	0.0248
0.466884	14.023	93.45	0.11×10^{-2}	0.472×10^{-2}	0.123×10^{-5}	0.115×10^{-2}	0.959×10^{-4}	0.285×10^{-4}	0.0660
0.454220	10.99	91.32	0.64×10^{-3}	0.35×10^{-2}	0.66×10^{-6}	0.21×10^{-2}	0.76×10^{-4}	0.338×10^{-4}	0.4244
0.000000	1.5196	34.20	0.29×10^{-4}	0.495×10^{-4}	0.19×10^{-8}	0.171	0.226×10^{-4}	0.232×10^{-4}	0.8595
0.000000	1.1395	12.24	0.46×10^{-5}	0.52×10^{-5}	0.296×10^{-10}	0.612×10^{-1}	0.81×10^{-6}	0.243×10^{-6}	0.8882
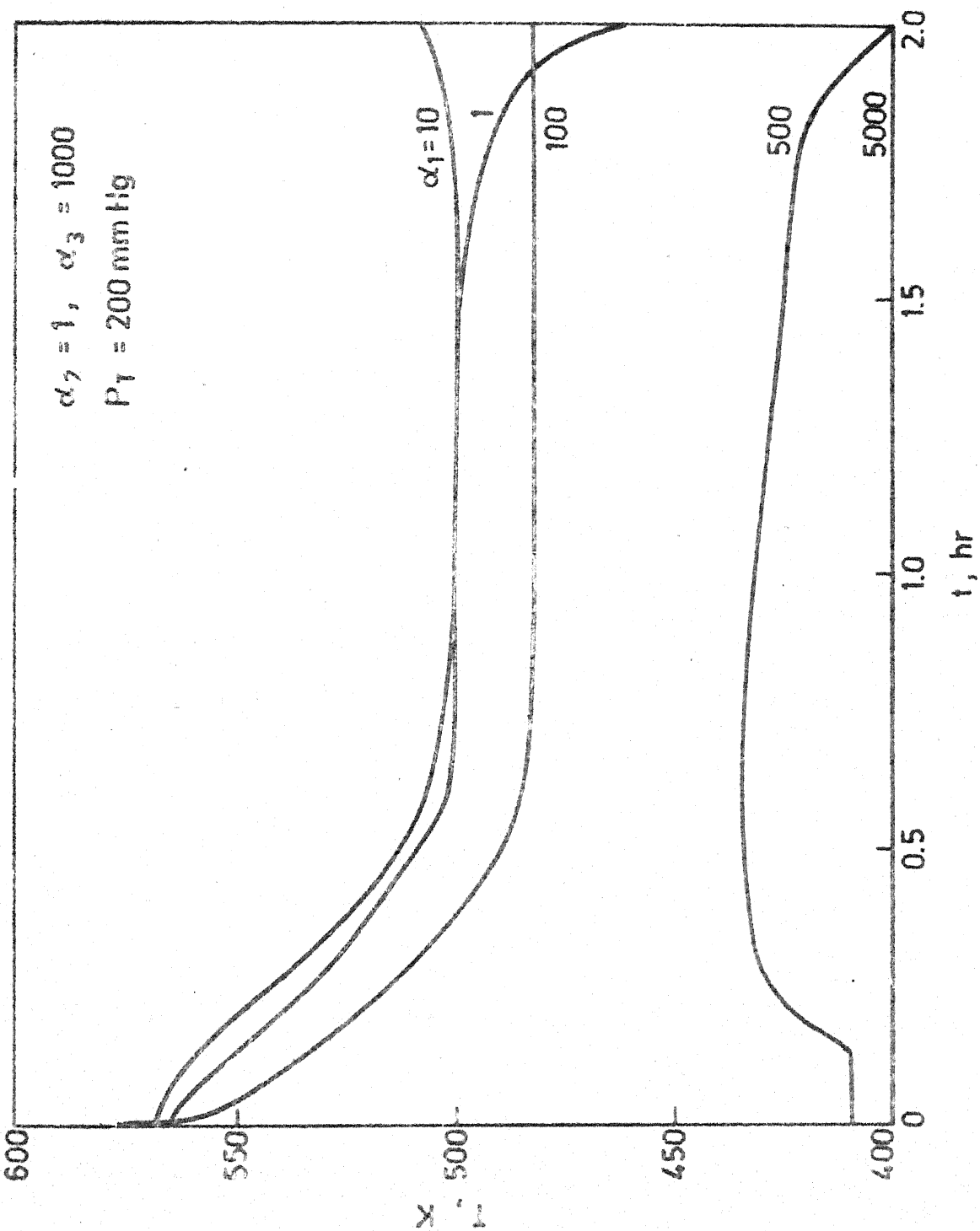
Fig. 7 Effect of variation of α_1 on the optimal temperature profile.
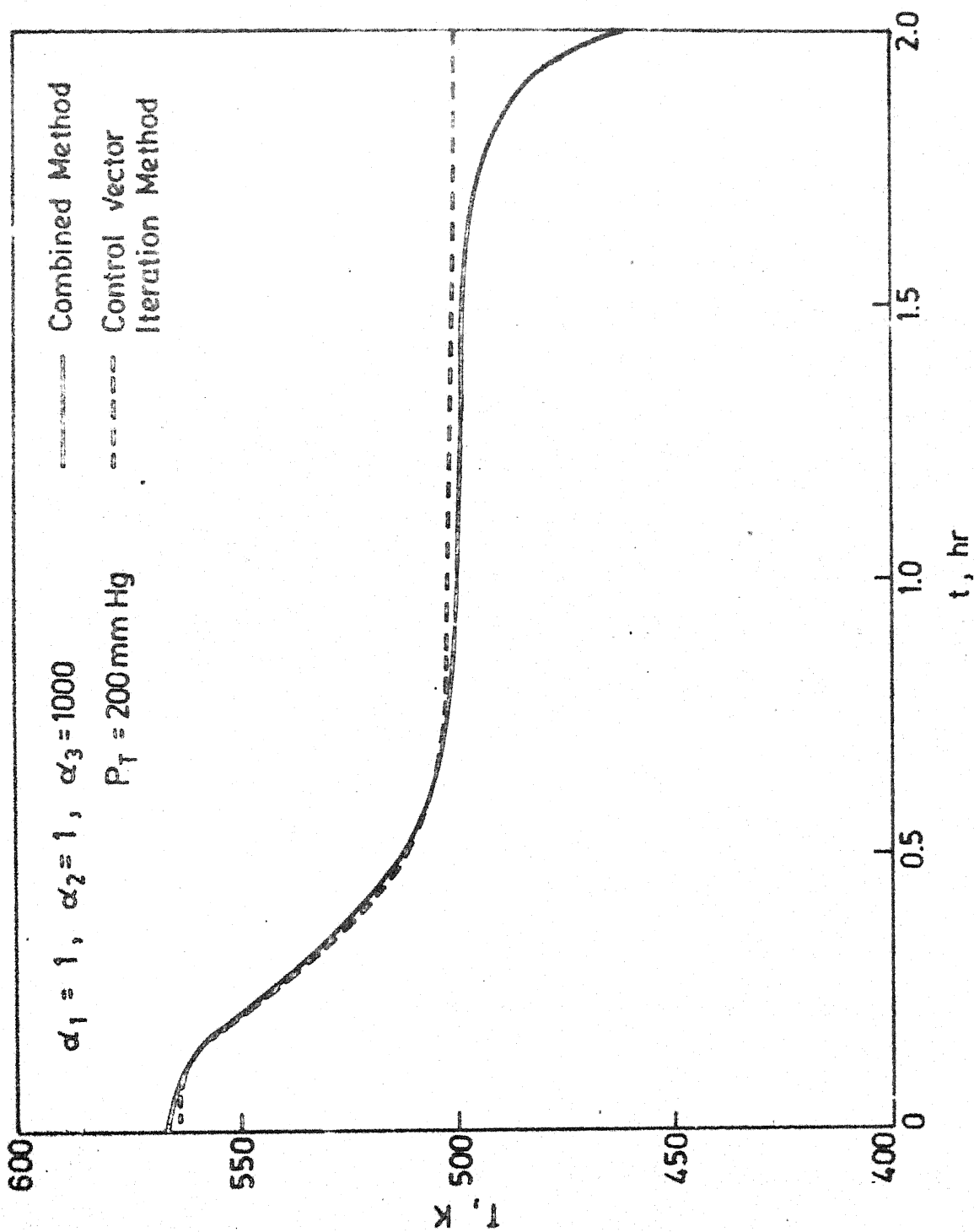
Fig. 8 Comparison of the optimal temperature profiles obtained by the combined method and the control vector iteration method.

71

because the optimization problem reduces completely to the minimization of the first term in the objective function. In Fig.8, the optimal temperature profiles obtained using the two methods are compared for a particular value of α_1 and the same behaviour is encountered as in Fig.5.

The weighting parameter α_3 represents the relative importance of the formation of side products in the reaction mass. By varying α_3, the results obtained are given in Table 11 and Fig. 9. In Table 11, it is seen that as α_3 increases, the conversion of Eg and the value of μ_n both increases, but the increase in these values is less when compared to the corresponding values in Table 9, when α_2 is varied. This shows that as α_3 is increased, the optimization problem reduces to the minimization of side products, thereby bringing down the conversion and μ_n to low values.

In Fig.9 as α_3 increases from 1 to 10^5, the first three curves show that the initial temperatures are low, they increase to high values after sometime and then are brought down to low values towards the end. This behaviour is peculiar to that expected since an increase in α_3 should lead to a reduction in the reactor temperature as more weightage is given to the minimization of side products. This peculiar behaviour represents a complex interplay of the three terms in the objective function.

TABLE 11

EFFECT OF VARIATION OF α_3 ON THE VARIOUS RESULTS

$\alpha_1 = 1000$

$\alpha_2 = 1$

$P_T = 200$ mmHg

Amount flashed	μ_n	Conversion,%	e_c	d	e_v	g	w	a	I
1.0 0.0000	1.174	14.83	0.64×10^{-5}	0.747×10^{-5}	$0.578\ 10^{-10}$	0.742×10^{-1}	0.141×10^{-5}	0.348×10^{-6}	0.8673
.0 0.0000	1.155	13.46	0.5416×10^{-5}	0.62×10^{-5}	0.393×10^{-10}	0.673×10^{-1}	0.108×10^{-5}	0.288×10^{-6}	0.8669
.0 0.2726	2.800	64.42	0.304×10^{-3}	0.943×10^{-3}	$.195 \times 10^{-6}$	0.491×10^{-1}	0.199×10^{-3}	0.105×10^{-4}	1.6188
.) 0.3764	9.306	89.55	0.408×10^{-3}	0.257×10^{-2}	0.455×10^{-5}	0.305×10^{-2}	0.588×10^{-4}	0.817×10^{-5}	2.5820

Fig. 9 Effect of variation of α_3 on the optimal temperature profile.

$\alpha_1 = 1000$, $\alpha_2 = 1$

$P_T = 200$ mm Hg

$\alpha_3 = 10^3$

10^5

10^4

1
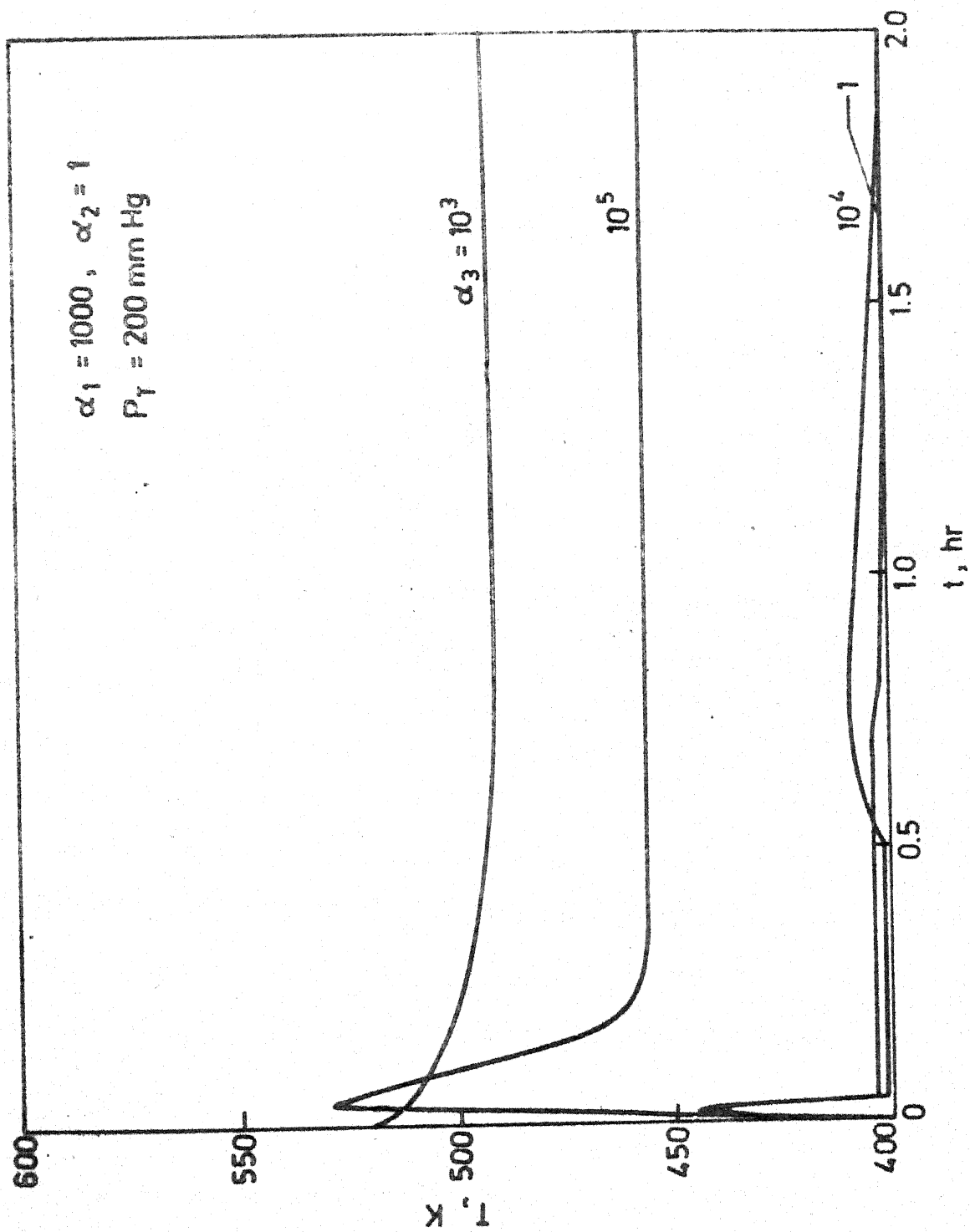
The effect of the pressure on the reactor performance
has been studied and the results obtained are given in Table 12
and Fig.10. It is found from Table 12 that as the pressure is
reduced, the conversion of E_g and the value of μ_n increases.
This behaviour is expected since the ~~diffusion~~ of the conden-
sation product is easier when the pressure is reduced, thereby
favouring the polymerization reaction. It is found from Fig.10
that the initial temperatures are low, when the pressure is
reduced from 200 to 20 mm Hg, but for low pressure, the tempera-
tures increase after sometime and then begin to fall towards the
end. This behaviour is entirely different from that predicted
by Eq.(4) in which the temperature is an increasing function
of time. In Fig. 11 the optimal temperature profiles for the
two methods are compared for a reactor pressure of 50 mmHg
and it is seen that there is a considerable change in the two
profiles.

In Tables 13 and 14, the objective functions obtained from
the control vector iteration method and the second variation
method are compared and it can be seen that there is only small
reduction in the value of the objective function obtained from
the latter method. This is expected since the second variation
method, derived by considering small perturbations in the
value of the state and control variables around the optimal
profile obtained from the control vector iteration method
incorporates only small changes in the objective function
obtained from the latter method.

TABLE 12

EFFECT OF VARIATION OF P_T ON THE VARIOUS RESULTS

$\alpha_1 = 1000$ $\alpha_2 = 5$ $\alpha_3 = 1000$

P_T mm Hg	Amount flashed	μ_n	Conversion,%	e_c	d	e_v	g	w	a	I
200	0.4259	7.141	86.3	0.49×10^{-3}	0.255×10^{-2}	0.516×10^{-6}	0.542×10^{-2}	0.968×10^{-4}	0.232×10^{-4}	3.9547
150	0.4579	11.938	91.97	0.49×10^{-3}	0.294×10^{-2}	0.631×10^{-6}	0.175×10^{-2}	0.534×10^{-4}	0.257×10^{-4}	3.2108
50	0.4731	13.588	92.46	0.12×10^{-3}	0.104×10^{-2}	0.592×10^{-6}	0.14×10^{-2}	0.106×10^{-4}	0.185×10^{-6}	1.5196
20	0.4829	23.897	95.93	0.169×10^{-4}	0.108×10^{-2}	0.642×10^{-6}	0.293×10^{-3}	0.107×10^{-6}	0.167×10^{-8}	1.5864
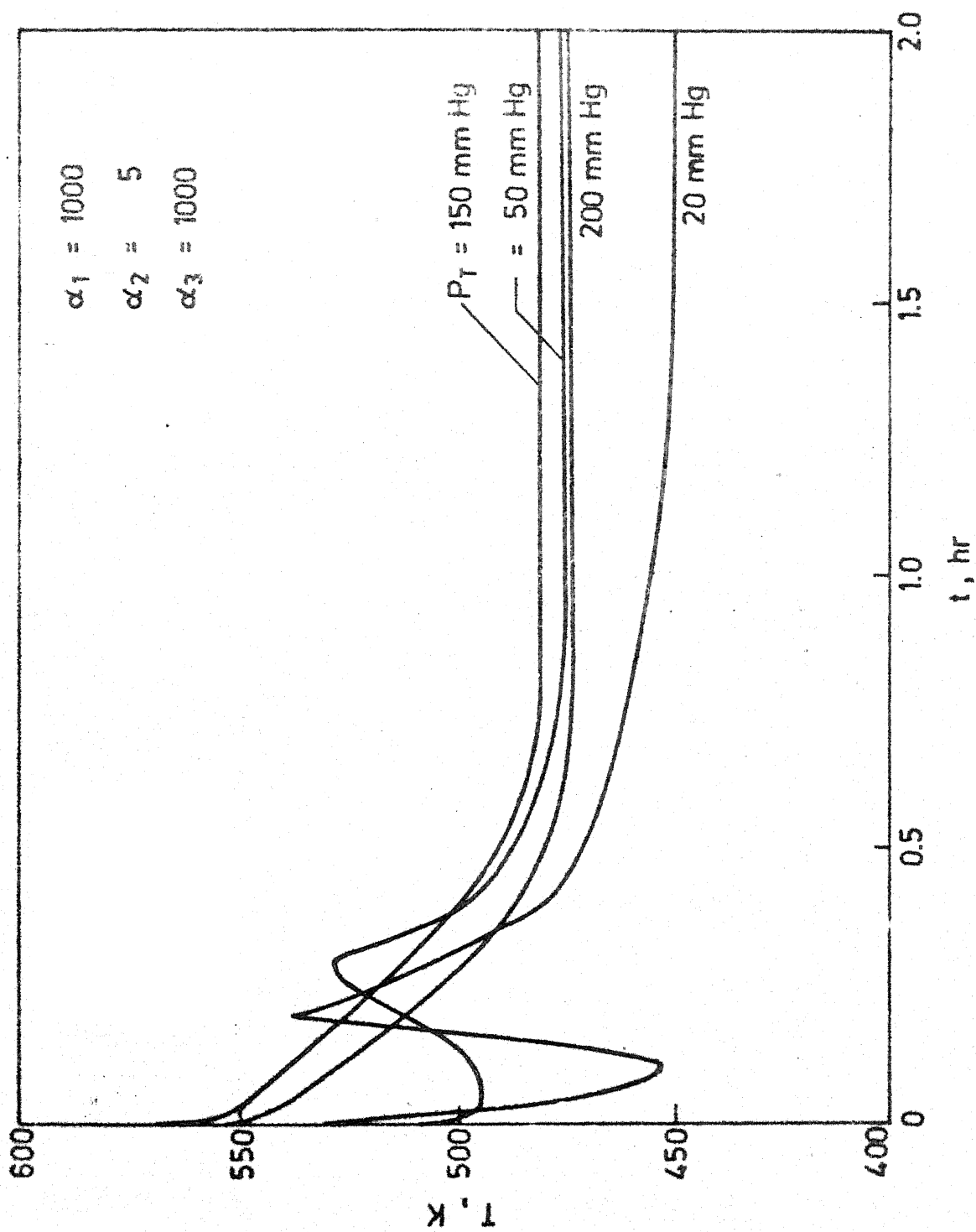
Fig. 10 Effect of variation of P_T on the optimal temperature profile.
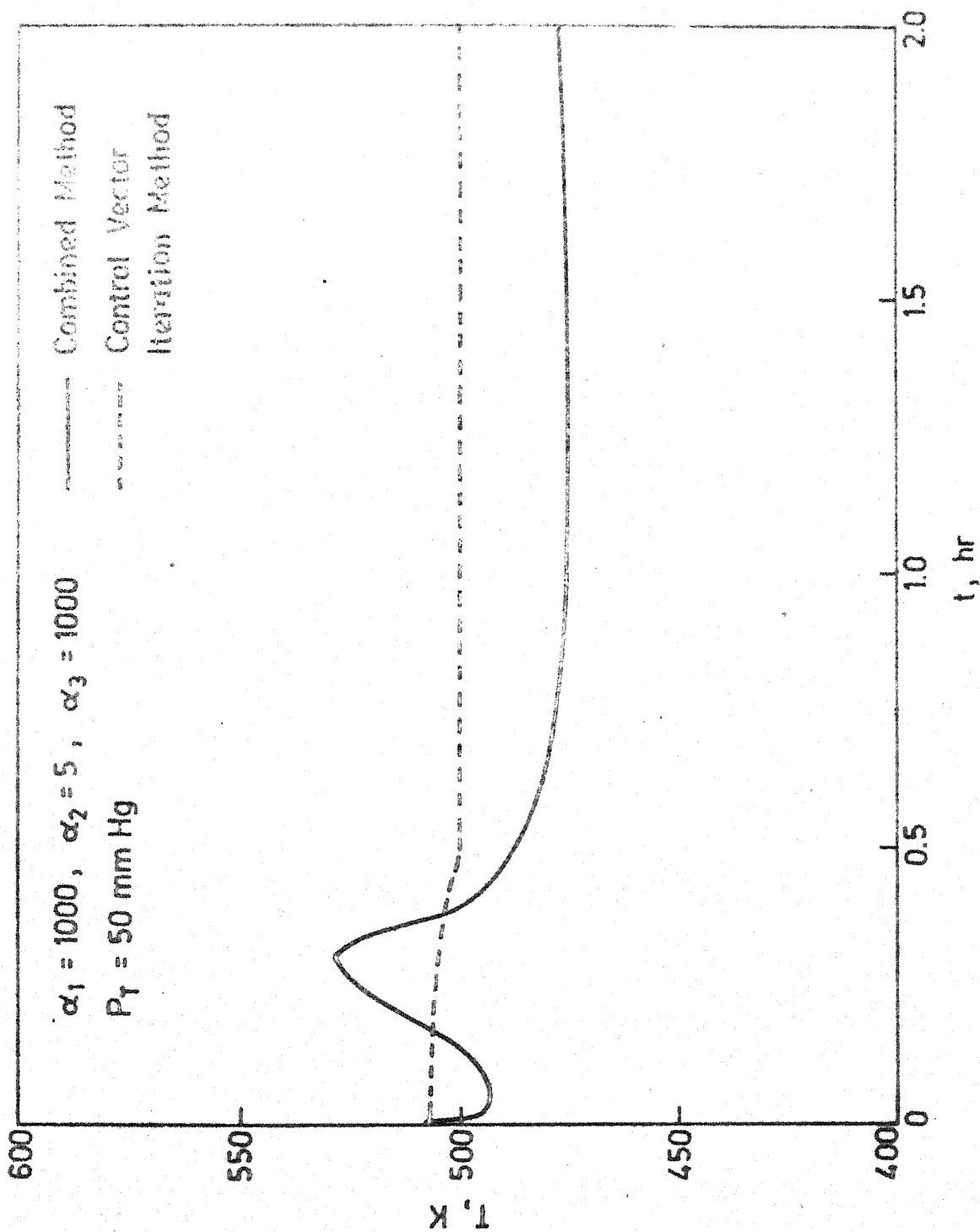
Fig. 11 Comparison of the optimal temperature profiles obtained by the combined method and the control vector iteration method.

TABLE 13

COMPARISON OF THE OBJECTIVE FUNCTIONS OBTAINED FROM THE CONTROL VECTOR ITERATION METHOD AND THE COMBINED METHOD (i) VARYING α_1 AND (ii) VARYING α_2

$\alpha_1 = 1000 \quad \alpha_3 = 1000 \quad P_T = 200 \text{ mmHg}$

α_2	Control vector iteration method I	Combined method I	Time at which $P \to \infty$, hr	Iteration number
0.5	0.4363	0.4363	–	–
1.0	0.8667	0.8667	0.02	4
5.0	3.9547	3.9547	1.88	3
10.0	4.2653	4.2653	1.96	5
100.0	6.1701	6.0348	–	–
1000.0	21.2972	21.0261	0.1	4

$\alpha_2 = 1 \quad \alpha_3 = 1000 \quad P_T = 200 \text{ mmHg}$

α_2	Control vector iteration method I	Combined method I	Time at which $P \to \infty$, hr	Iteration number
1.0	0.054	0.0248	0.14	5
10.0	0.0897	0.066	0.08	4
100.0	0.4244	0.4244	1.96	3
500.0	0.8595	0.8595	1.96	3
5000.0	0.8882	0.8882	–	–

TABLE 14

COMPARISON OF THE OBJECTIVE FUNCTIONS OBTAINED FROM THE CONTROL VECTOR ITERATION
METHOD AND THE COMBINED METHOD (i) VARYING α_3 AND (ii) VARYING P_T

(i) $\alpha_1 = 1000$ $\alpha_2 = 1$ $P_T = 200$ mmHg

α_3	Control vector iteration method I	Combined method I	Time at which $P \to \infty$, hr	Iteration number
1.0	0.8673	0.8673	0.02	4
0000.0	0.8670	0.8669	0.02	4
00000.0	1.6209	1.6188	0.02	4
000000	2.6989	2.5820	—	—

(ii) $\alpha_1 = 1000$ $\alpha_2 = 5$ $\alpha_3 = 1000$

P_T mmHg	Control vector iteration method I	Combined method I	Time at which $P \to \infty$, hr	Iteration Number
200	3.9547	3.9547	1.88	3
450	3.2108	3.2108	1.88	3
50	1.8737	1.5196	—	—
20	1.9544	1.5864	—	—

It can be seen from Eq.(36) that the correction to be employed in   the temperature profile depends upon the value of ε. It is found that the convergence is fast i.e., a maximum decrease in the objective function is obtained for a value of ε =1 for low pressures, whereas for high pressures, convergence is attained for relatively low values of ε.

All the results are generated using DEC-1090 computer and it takes about 30 minutes of computation time for obtaining the optimal temperature profile using the combined method.

CONCLUSIONS

The polycondensation stage of PET formation has been
assumed to include various side reactions in addition to the
usual polymerization reaction. A simplified model for the
flashing process is proposed in which it is assumed that the
flashing occurs at the end of small discrete time intervals.
During this time, polymerization is assumed to occur in batch
reactor. An objective function proposed earlier has been
used in which there are three weighting parameters, α_1, α_2,
and α_3. A suitable Hamiltonian is written for the time
interval when the polymerization proceeds without flashing and
the corresponding adjoint equations have been written assuming
temperature and pressure as the control variables. While
computing the optimal temperature profiles, it was discovered
no matter what the values of the parameters α_1, α_2, and α_3
are chosen, pressure always falls to the lowest limit. This
leads to the simplification of the problem and in the following
we have treated temperature as the **only** control variable
for lower limits of pressure.

The optimal temperature profiles for a given lower limit
of pressure were obtained using a combined method of the
first and the second order techniques of optimization. In the
first order technique, the control vector iteration method is
used till the change in the objective function is negligibly

small. It is well known that the convergence in first variation technique slows down considerably as the optimality is approached. Because of this reason, the near optimal profiles obtained from the first variation technique is used as the initial guess for the subsequent second variation technique. The solution of the Riccati equation in the second variation method poses both a memory storage as well as computational problem. The former has been overcome by storing the values at every time interval of 10^{-2} hr. For obtaining stable numerical solutions for the Riccati equation, the integration has been carried out using a time interval of 2×10^{-4} hr and obtaining the intermediate values of temperature, state, and adjoint variables by linear interpolation. In this integration sometimes, the Riccati variables, **P**, approached infinity after some time of integration. Bryson and Ho suggests that is because of the existence of conjugate paths. Therefore, we stopped the integration at this point and assumed **P** to be zero for lesser times. This would lead to the correction of temperature profile for that portion of time, where the values of **P** are finite. In this way, we could obtain stable numerical solutions. The weighting parameters involved in the objective function proposed for minimization, were found to have considerable influence on the optimal temperature profile we found that the combined method works better for obtaining optimal solutions rather than each method used separately.

Computations suggest one must use high temperatures initially
which should be lowered subsequently. It has been suggested
that the high temperatures initially could be attained in
practice by using PET of high molecular weight as an inert.
We also found that the optimal temperature profiles were found
to depend significantly upon the reactor pressure and it was
found that for low pressures, one must increase the temperature
during the earlier portion of polymerization time which once
again be lowered for large times to reduce the formation of side
products.

REFERENCES

1. M. Katz in Polymerization Processes, C.E.Schildknecht and I. Skeist, eds. Wiley Interscience, N.Y., 1971.

2. S.K. Gupta and A. Kumar, Chem. Eng.Commun., 20, 1 (1983).

3. K. Ravindranath and Mashelkar, R., Chem. Eng. Sci., to be published.

4. J.W. Ault and D.A. Mellichamp, Chem. Eng.Sci., 27, 2219 (1972).

5. K. Ravindranath and R.A. Mashelkar, J.Appl. Polym. Sci., 26, 3179 (1981).

6. A. Arunpal and A. Vineypal, Chem.Tech., 9, 260 (1979).

7. T.Tani and K. Enoki, Hydrocarbon Proc., 49, 146 (1970).

8. K. Ravindranath and R.A. Mashelkar, J. Appl. Polym.Sci., 27, 2625 (1982).

9. A. Kumar, S.K. Gupta, B. Gupta, and D. Kunzru, J. Appl. Polym. Sci., 27, 4421 (1982).

10. A. Kumar, S.K. Gupta, and N. Somu, Polym. Eng. Sci., 22, 314 (1982).

11. A. Kumar, S.K. Gupta, M.V.S. Rao and N.Somu, Polymer, in press. 24, 669 1983

12. P. Ellwood, Chem. Engg., Nov. 20, 98 (1967).

13. R.M. Secor, AIChE J., 15, 861 (1969).

14. M.Amon and C.D. Denson, I & E.C. Fundam., 19, 415 (1980).

15. S.K. Gupta,A. Kumar, and A.Ghosh, I. & E.C. Fundam., 22, 268 (1983).

16. A. Kumar, S.K. Gupta, S. Madan, N.G. Shah, and S.K. Gupta, Polym.Eng.Sci., 24, 194 (1984).

17. S.K. Gupta, A.K. Ghosh, S.K. Gupta, and A. Kumar, J. Appl. Polym.Sci., in press.

18. V.V. Chavan and R.A. Mashelkar, Advances in Transport Processes, Vol.1, P.210, 1980.

19. Li-Chen Hsu, J. Macromol. Sci., Physics, B1(4), 801 (1967).

20. L.H. Buxhaum, Angew Chem. Int. Ed., 7(3), 182 (1968).

21. K. Yoda, K. Kimoto and J. Toda, J. Chem. Soc. Japan (Ind. Chem. Sect.) 67, 907 (1967).

22. S.G. Hovenkamp, J. Polym.Sci., A-1, 2, 3617 (1971).

23. C.C. Walker, J. Polym.Sci. Polym. Chem. Ed., 21, 623 (1983).

24. G. Tsimmerman and E.Shaaf, Vysokomol Soyed Ser. A, 15, 415 (1973).

25. G. Rafler, G. Reinisch, and E. Bonatz, Acta Chimica (Budapest), 81, 253 (1974).

26. H. Kamatani, S. Konagaya, and Y. Nakamura, Polymer, 12, 125 (1980).

27. H. Kamatani, S. Konagaya, Konbunshi Ronbunshu, 35, 787 (1978).

28. R.W. Stevenson, J. Polym.Sci., 7, 395 (1969).

29. K. Tomita, Polymer, 17, 221 (1976).

30. K. Tomita, Konbunshi Ronbunshu, 33, 96 (1976).

31. A. Kumar, V.K. Sukthankar, C.P. Vaz, and S.K. Gupta, Polym. Eng.Sci., 24, 185 (1984).

32. A. Kumar, S.N. Sharma, and S.K. Gupta, J. Appl. Polym. Sci., in press.

33. G. Challa, Macromol. Chem., $\underline{38}$, 105, 123, 138 (1960).

34. S.K. Gupta, B.Damania, and A. Kumar, J. Appl. Polym. Sci., in Press.

35. R.C. Reid, J.M. Prausnitz, and T.K. Sherwood, The Properties of Gases and Liquids, 3rd ed., McGraw Hill, N.Y., 1977.

36. W.H. Ray and J. Szekely, Process Optimization, 1st ed., Wiley,N.Y., 1973.

37. Leon Lapidus and Rein Luus, Optimal Control of Engineering Processes, Blaisdell, Waltham, 1967.

38. Morton M.Denn, Optimization by Variational Methods, McGraw Hill, N.Y., 1969.

39. C.W. Merriam, Optimization Theory and the Design of Feedback Control Systems, McGraw Hill, N.Y., 1964.

40. Arthur E. Bryson, Jr. and Yu-Chi Ho, Applied Optimal Control, Waltham, 1969.

APPENDIX

1. COMPUTER PROGRAM FOR THE CONTROL VECTOR ITERATION METHOD

```
******************************************************************
      OPTIMIZATION OF PET REACTORS                              *
******************************************************************

      THIS PROGRAM USES THE CONTROL VECTOR ITERATION METHOD OR THE
      GRADIENT METHOD FOR FINDING THE OPTIMAL TEMPERATURE PROFILES
      IN A BATCH PET REACTOR

      THIS METHOD IS CARRIED OUT TILL THE CHANGE IN THE OBJECTIVE
      FUNCTION FROM ITERATION TO ITERATION IS SMALL

      ISOTHERMAL TEMPERATURE IS USED FOR THE INITIAL GUESS
******************************************************************

                  MAIN PROGRAM

******************************************************************

      X          STATE VARIABLES
      V          ADJOINT VARIABLES
      E0         ACTIVATION ENERGY
      Z0         FREQUENCY FACTOR
      Z          RATE CONSTANTS FOR THE REACTIONS INVOLVED IN THE POLY-
                 CONDENSATION STAGE OF PET FORMATION
      A          WIDTH OF THE TIME INTERVAL USED IN INTEGRATIONS WITH A
                 TOTAL TIME OF 2 HRS.
      EG0        INITIAL CONCENTRATION OF EG GROUP IN THE FEED
      PT         TOTAL PRESSURE,ATM.
      T          REACTOR TEMPERATURE,DEGREE KELVIN
      TNEW       TEMPERATURE AT 101 POINTS OF TIME
      TN         TEMPERATURE AT 2001 POINTS OF TIME
      R          UNIVERSAL GAS CONSTANT
      ZP         DERIVATIVE OF Z W.R.T. T
      VDMT       MOLAR VOLUME OF DMT
      VG         MOLAR VOLUME OF GLYCOL
      VW         MOLAR VOLUME OF WATER
      VT         TOTAL VOLUME OF THE REACTION MASS,LITRES
      Z1         Z DIVIDED BY VT
      HP         DERIVATIVE OF HAMILTONIAN W.R.T. TEMP
      PI         OBJECTIVE FUNCTION
      VP         DERIVATIVE OF VT W.R.T. T
      E          EPSILON-FACTOR DETERMINING THE MAGNITUDE OF STEP TO BE
                 TAKEN IN THE CORRECTION OF TEMPERATURE FROM THE PREVI-
                 OUS ITERATON
      EOPT       OPTIMUM EPSILON
      AK         VAPOUR PRESSURE,ATM
                 ALPHA1 - WEIGHTING PARAMETER FOR THE DESIRED CONCENTR-
                 ATION OF DIETHYLENE GLYCOL IN THE POLYMER
                 ALPHA2 - WEIGHTING PARAMETER FOR THE DESIRED VALUE OF
                 DPBAR OF THE POLYMER
                 ALPHA3 - WEIGHTING PARAMETER FOR THE FORMATION OF
                 SIDE PRODUCTS IN THE REACTON MASS
      DPBAR      NUMBER-AVERAGE MOLECULAR WEIGHT OF THE POLYMER

******************************************************************

      DIMENSION X(7,2001),V(4,2001),Z0(9),E0(9)
      DIMENSION T(20),TNEW(101),TR(101),TN(2001)
      DIMENSION Z(9,2001),ZP(9,101),HP(101),EP(15),QI(15)
```

```
      DIMENSION VDMT(2001),VG(2001),VW(2001),VT(2001)
      DIMENSION E3(10),F3(10),Z1(9,2001),DPBAR(2001),OF(15)
**********************************************************************
      COMMON/A1/SSX(6)/A2/X/A3/Z
      COMMON/A4/VW,VDMT,VG,VT/E1/EO,ZO,R
      COMMON/A5/Z1/A6/TN/A7/V
      COMMON/A8/ZP,HP
      COMMON/A9/EGO/A30/TNEW
      COMMON/FF1/PT,H
      COMMON/AP/DISTF
      COMMON/A12/DPBAR
**********************************************************************
      REAL 4P
      PNEW=0.0
      EGO=1.0
      H=-1.0
      TYPE 540
      FORMAT(1X,'GIVE THE VALUES OF ALPHA1,ALPHA2,ALPHA3 IN D.LINES')
      READ(5,*),ALPHA1,ALPHA2,ALPHA3
      WRITE(60,3),ALPHA1,ALPHA2,ALPHA3
      FORMAT(1X,'ALPHA1=',F15.5,2X,'ALPHA2=',F15.5,2X,'ALPHA3=',F15.5)
      H=101
      TYPE 673
      FORMAT(1X,'GIVE TOTAL PRESSURE')
      READ(5,*),PT
      PI=0.0
      A=1.0/1000.0
      R=1.987
      NO=0

      READ(24,*),(TN(I),I=1,11)
      WRITE(63,1199),TN(1),PT
      FORMAT(1X,'GUESS TEMP:=',F10.3,'TOTAL PRESSURE=',F15.11)
      OPEN(UNIT=23)
      TNEW(1)=TN(1)
      J1=2
      DO 444 J=1,10
      D=1.0
      DO 444 I=1,10
      T(I)=(TN(J+1)-TN(J))*D/10.0+TN(J)
      TNEW(J1)=T(I)
      D=D+1.0
      J1=J1+1
      CONTINUE
      EO(3)=18500.00
      EO(4)=29800.00
      EO(5)=29800.00
      EO(6)=29800.00
      EO(7)=17600.00
      EO(8)=17600.00
      EO(9)=37800.00
      ZO(3)=13.43
      ZO(4)=17.54
      ZO(5)=17.54
      ZO(6)=17.54
      ZO(7)=13.855
      ZO(8)=13.855
      ZO(9)=22.004
      NO=NO+1
      WRITE(60,16)
      WRITE(60,12),NO
```

```
      FORMAT(1X,120(1H-))
      TYPE 12,NO
      FORMAT(1X,'NO=',I2)
      WRITE(60,16)
      WRITE(60,16)
      CALCULATION OF MOLAR VOLUMES AND RATE CONSTANTS AT 2001
      POINTS OF TIME
      DO 19 I=3,9
      Z(I,1)=60.0*EXP(-EO(I)/(R*TNEW(1))+ZO(I))
      ZP(I,1)=EO(I)*Z(I,1)/(R*TNEW(1)*TNEW(1))
      CONTINUE
      TN(1)=TNEW(1)
      VDMT(1)=191.5*(1.0+0.0014*(TN(1)-413.0))/1000.0
      VG(1)=60.6*(1.0+0.0014*(TN(1)-413.0))/1000.0
      VW(1)=(19.422+0.025*(TN(1)-413.0))/1000.0
      J1=2
      DO 14 J=1,100
      D=1.0
      DO 14 I=1,20
      T(I)=(TNEW(J+1)-TNEW(J))*D/20.0+TNEW(J)
      DO 15 I1=3,9
      Z(I1,J1)=60.0*EXP(-EO(I1)/(R*T(I))+ZO(I1))
      CONTINUE
      TN(J1)=T(I)
      VDMT(J1)=191.5*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VG(J1)=60.6*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VW(J1)=(19.422+0.025*(TN(J1)-413.0))/1000.0
      D=D+1.0
      J1=J1+1
      CONTINUE
      INITIAL CONDITIONS ON THE STATE VARIABLES
      X(1,1)=1.0
      X(2,1)=0.0
      X(3,1)=0.0
      X(4,1)=0.0
      X(5,1)=0.0
      X(6,1)=0.0
      X(7,1)=0.0
      CALL RUNG1(A)
      WRITE(60,678),((X(I,J),J=1,2001,100),I=1,7)
      FORMAT(2X,'X IS: ',21F10.3)
      WRITE(60,16)
      TYPE 791,NO,DISTF
      FORMAT(1X,'FOR NO=',I2,'AMOUNT FLASHED=',F15.8)
      WRITE(63,791),NO,DISTF
      DISTF=0.0
      FINAL CONDITIONS ON THE ADJOINT VARIABLES
      V(1,2001)=0.0
      V(2,2001)=0.0
      V(3,2001)=ALPHA1
      V(4,2001)=0.0
      MP=(X(1,2001)+X(2,2001)+X(3,2001)+X(4,2001))/2.0
      VT(2001)=VDMT(2001)*MP+VG(2001)*X(5,2001)+VW(2001)*X(6,2001)
      Z1(4,2001)=Z(4,2001)
      Z1(9,2001)=Z(9,2001)
      DO 191 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 191
      Z1(I,2001)=(Z(I,2001)/VT(2001))
      CONTINUE
      CALL RUNG2(ALPHA2,ALPHA3,A)
      WRITE(63,600),((V(I,J),J=1,2001,100),I=1,4)
```

```
        WRITE(63,16)
        TYPE 12,NO
500     FORMAT(2X,'V IS: ',21E11.3)
        PPI=PI
        CALL SIMPSN
        PI=ALPHA1*X(3,2001)+(ALPHA2/225.0)*SSX(1)+ALPHA3*(SSX(2)
       1+SSX(4))
        OF(NO)=PI
        TYPE 28,PI
        WRITE(60,28),PI
28      FORMAT(1X,'PI=',F20.5)
        TYPE 113
        READ(5,455),C1
        IF(C1.EQ.'N')GOTO 42
        IF(NO.EQ.1)GOTO 1516
        IF((ABS(OF(NO-1)-OF(NO))).LE.0.001)GOTO 42
1516    IF(NO.GT.8) GOTO 42
        DO 32 J=1,100
        DO 30 I=1,7
        X(I,J+1)=X(I,J*20+1)
        IF(I.GT.4)GOTO 30
        V(I,J+1)=V(I,J*20+1)
30      CONTINUE
        VT(J+1)=VT(J*20+1)
32      CONTINUE
        DO 33 I=3,9
        IF((I.EQ.4).OR.(I.EQ.9))GOTO 33
        MP=(X(1,1)+X(2,1)+X(3,1)+X(4,1))/2.0
        VP=(((MP)*191.5*0.0014+X(5,1)*60.6*0.0014+X(6,1)*0.025)
       1/1000.0)
        ZP(I,1)=(ZP(I,1)/VT(1))-((Z(I,1)/(VT(1)*VT(1)))*VP)
33      CONTINUE
        DO 31 J=1,100
        DO 31 I=3,9
        Z(I,J+1)=60.0*EXP(-EO(I)/(R*TNEW(J+1))+ZO(I))
        ZP(I,J+1)=EO(I)*Z(I,J+1)/(R*TNEW(J+1)*TNEW(J+1))
        IF((I.EQ.4).OR.(I.EQ.9))GOTO 31
        MP=(X(1,J+1)+X(2,J+1)+X(3,J+1)+X(4,J+1))/2.0
        VP=(((MP)*191.5*0.0014+X(5,J+1)*60.6*0.0014+X(6,J+1)*0.025)
       1/1000.0)
        ZP(I,J+1)=(ZP(I,J+1)/VT(J+1))-((Z(I,J+1)/(VT(J+1)*VT(J+1)))*VP)
31      CONTINUE
        CALL DHU(M)
        TYPE 22,(HP(J),J=1,M)
22      FORMAT(1X,'THE VALUE OF HP IS',101E15.4)
        WRITE(60,22),(HP(J),J=1,M)
        TYPE 12,NO
18      FORMAT(1X,'THE STARTING EPSILON=',F15.4,'IN THE INTERVAL=',F15.4
       1)
C       THIS PART CALCULATES OPTIMUM EPSILON
C****************************************************************
121     TYPE 29
        DO 601 K2=1,10
        EP(K2)=0.0
        QI(K2)=0.0
601     CONTINUE
        K1=1
29      FORMAT(1X,'GIVE THE VALUE OF STARTING EPSILON AND INCREMENT.')
        READ(5,*),E,FNEW
        E3(NO)=E
        F3(NO)=FNEW
```

```
        WRITE(60,7003),F3(NO),F3(NO)
7003    FORMAT(1X,'THE STARTING EPSILON=',F15.5,'THE INCREMENT=',F15.5)
1007    FORMAT(F30.10)

57      DO 51 J=1,101
        TR(J)=TNEW(J)-E*HP(J)
        IF(TR(J).LT.600.0)GOTO 5
        TR(J)=600.0
5       IF(TR(J).GT.400.0) GOTO 51
        TR(J)=400.0
51      CONTINUE
        DO 60 I=3,9
        Z(I,1)=60.0*EXP(-EO(I)/(R*TR(1))+ZO(I))
60      CONTINUE
        VDMT(1)=191.5*(1.0+0.0014*(TR(1)-413.0))/1000.0
        VG(1)=60.6*(1.0+0.0014*(TR(1)-413.0))/1000.0
        VW(1)=(19.422+0.025*(TR(1)-413.0))/1000.0
        J1=2
        TN(1)=TR(1)

        DO 52 J=1,100
        D=1.0
        DO 52 I=1,20
        T(I)=(TR(J+1)-TR(J))*D/20.0+TR(J)
        DO 53 I1=3,9
        Z(I1,J1)=60.0*EXP(-EO(I1)/(R*T(I))+ZO(I1))
53      CONTINUE
        TN(J1)=T(I)
        VDMT(J1)=191.5*(1.0+0.0014*(TN(J1)-413.0))/1000.0
        VG(J1)=60.6*(1.0+0.0014*(TN(J1)-413.0))/1000.0
        VW(J1)=(19.422+0.025*(TN(J1)-413.0))/1000.0
        D=D+1.0
        J1=J1+1
52      CONTINUE
        CALL RUNG1(A)
        DISTF=0.0
        TYPE 166,K1
166     FORMAT(1X,'NO.INSIDE THE LOOP=',I2)
        TYPE 678,(X(1,J),J=1,2001,100)
        CALL SIMPSN
        PII=ALPHA1*X(3,2001)+(ALPHA2/225.0)*SSX(1)+ALPHA3*(SSX(2)
     1+SSX(4))
        EP(K1)=E
        TYPE 82
82      FORMAT(1X,'MORE?')
        READ(5,455),B1
        IF(B1.EQ.'N')GOTO 72
        OT(K1)=PII
        TYPE 71,EP(K1),OI(K1)
        TYPE 82
        READ(5,455),B1
        IF(B1.EQ.'N')GOTO 72
        E=E+FNEW
        K1=K1+1
        IF(K1.LT.11)GOTO 57
72      TYPE 12,NO
        TYPE 123
123     FORMAT(1X,'DO YOU WANT TO GO BACK IN E.OPT.LOOP(Y/N)?')
        READ(5,455),H1
        IF(H1.EQ.'Y')GOTO 121
C*********************************************************************
```

```
          WRITE(60,16)
          TYPE 12,NO
          WRITE(60,65)
          TYPE 65
65        FORMAT(15X,'I',30X,'E')
          DO 70 K1=1,10
          WRITE(60,71),OI(K1),EP(K1)
          WRITE(5,71),OI(K1),EP(K1)
71        FORMAT(7X,F17.4,3X,F17.7)
70        CONTINUE
64        FORMAT(2X,'EOPT=',E20.10)
          TYPE 1008
1008      FORMAT(1X,'GIVE THE VALUE OF EOPT')
          READ(5,*),AL
1009      FORMAT(F15.4)
          EOPT=AL
          WRITE(60,16)
          WRITE(60,64),EOPT
C**********************************************************************
C         THIS PART CORRECTS THE OLD TEMP.PROFILE USING EOPT
          DO 40 J=1,M
          TNEW(J)=TNEW(J)-EOPT*HP(J)
          IF(TNEW(J).LT.600.0)GOTO 17
          TNEW(J)=600.0
17        IF(TNEW(J).GT.400.0)GOTO 40
          TNEW(J)=400.0
40        CONTINUE
C**********************************************************************
41        FORMAT(1X,I2)
44        FORMAT(1X,101F11.6)
          TYPE 45,(TNEW(J),J=1,M)
45        FORMAT(1X,'TNEW IS: ',101F7.2)
          WRITE(60,45),(TNEW(J),J=1,M)
113       FORMAT(1X,'DO YOU WANT TO PROCEED(Y/N)!')
455       FORMAT(A1)
          TYPE 113
          READ(5,455),F1
          IF(F1.EQ.'Y')GOTO 499
          GOTO 42
499       CONTINUE
49        GOTO 11
42        WRITE(23,549),(TNEW(J),J=1,M)
          DO 46 J=1,M
          IF(TNEW(J).LT.600.0)GOTO 4
          TNEW(J)=600.0
4         IF(TNEW(J).GT.400.0)GOTO 46
          TNEW(J)=400.0
46        CONTINUE
          TYPE 549,(TNEW(J),J=1,101)
549       FORMAT(1X,101F7.2)
646       FORMAT(1X,F7.2)
          WRITE(60,16)
          DPBAR(2001)=(EGO/(X(1,2001)+X(2,2001)+X(3,2001)+X(4,2001)))
          WRITE(59,899),(X(1,J),J=1,2001,20)
          WRITE(59,16)
          WRITE(59,899),(X(2,J),J=1,2001,20)
          WRITE(59,16)
          WRITE(59,899),(X(3,J),J=1,2001,20)
          WRITE(59,16)
          WRITE(59,899),(X(4,J),J=1,2001,20)
          WRITE(59,16)
```

```
      WRITE(59,899),(X(5,J),J=1,2001,20)
      WRITE(59,16)
      WRITE(59,899),(X(6,J),J=1,2001,20)
      WRITE(59,16)
      WRITE(59,899),(X(7,J),J=1,2001,20)
      WRITE(59,16)
      WRITE(59,899),(DPBAR(I),I=1,2001,20)
      WRITE(59,16)
899   FORMAT(1X,'X IS: ',101E15.6)
      WRITE(60,12),NO
      STOP
      END
C     *****************************************************************
C     THIS SUBROUTINE INTEGRATES THE STATE VARIABLE EQUATIONS
C     USING FOURTH-ORDER RUNGE-KUTTA METHOD
      SUBROUTINE RUNG1(A)
      DIMENSION X(7,2001),XORG(7),XI(7)
      DIMENSION XORG1(7),XR(7),TN(2001),EO(9),ZO(9)
      DIMENSION VDMT(2001),VG(2001),VW(2001),VT(2001)
      DIMENSION Z1(9,2001),ZI(9),DPBAR(2001),Z(9,2001)
      DIMENSION CP1(7),CP2(7),CP3(7),CP6(7),CP(7),FP(7)
C------------------------------------------------------------------
      COMMON/A2/X/A3/Z/A4/VW,VDMT,VG,VT/E1/EO,ZO,R
      COMMON/A5/Z1/A6/TN/RUNG11/FP,XR
      COMMON/R1/XI,XORG
      COMMON/A9/EGO
      COMMON/FF1/PT,H
      COMMON/R6/DISTF
      COMMON/A12/DPBAR
C------------------------------------------------------------------
      REAL MP
      REAL MPP
      A=A/5.0
      DO 100 J1=1,2000
      J=J1
      MP=(X(1,J)+X(2,J)+X(3,J)+X(4,J))/2.0
      DPBAR(J)=((EGO/MP)/2.0)
      VT(J)=VDMT(J)*MP+VG(J)*X(5,J)+VW(J)*X(6,J)
      DO 99 N=1,7
      XORG1(N)=X(N,J)
99    CONTINUE
      N=1
      D=0.0
      DO 12 I=3,9
      ZI(I)=Z(I,J)
12    CONTINUE
5555  CONTINUE
      DO 101 N=1,7
      XORG(N)=X(N,J)
      XR(N)=X(N,J)
101   CONTINUE
      T1=TN(J)+D*(TN(J+1)-TN(J))
      DO 1111 I=3,9
      Z(I,J)=60.0*EXP(-EO(I)/(R*T1)+ZO(I))
1111  CONTINUE
      X1=191.5*(1.0+0.0014*(T1-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T1-413.0))/1000.0
      X3=(19.422+0.025*(T1-413.0))/1000.0
      MPP=(XORG(1)+XORG(2)+XORG(3)+XORG(4))/2.0
      VTO=X1*MPP+X2*X(5,J)+X3*X(6,J)
      DO 2222 I=3,9
```

```
        IF((I.EQ.4).OR.(I.EQ.9))GOTO 2222
        Z1(I,J)=Z(I,J)/VTO
2222    CONTINUE
        Z1(4,J)=Z(4,J)
        Z1(9,J)=Z(9,J)
        D=D+0.2
        CALL EQTN1(J)
        DO 102 I=1,7
        CP1(I)=A*FP(I)
        XR(I)=XORG(I)+0.5*CP1(I)
102     CONTINUE
        CALL EQTN1(J)
        DO 103 I=1,7
        CP2(I)=A*FP(I)
        XR(I)=XORG(I)+0.5*CP2(I)
103     CONTINUE
        CALL EQTN1(J)
        DO 106 I=1,7
        CP3(I)=A*FP(I)
        XR(I)=XORG(I)+CP3(I)
106     CONTINUE
        CALL EQTN1(J)
        DO 105 I=1,7
        CP6(I)=A*FP(I)
        CP(I)=(CP1(I)+2.0*(CP2(I)+CP3(I))+CP6(I))/6.0
        X(I,J)=XORG(I)+CP(I)
105     CONTINUE
        N=N+1
        IF(N.LE.5) GOTO 5555
        DO 999 I=1,7
        X(I,J+1)=X(I,J)
        X(I,J)=XORG1(I)
999     CONTINUE
        DO 578 I=3,9
        Z(I,J)=ZI(I)
578     CONTINUE
        T=TN(J)
        IF(X(5,J+1).GT.0.0) GOTO 34982
        A1=Z1(3,J)*XORG(1)*XORG(1)+(Z1(7,J)*XORG(1)*XORG(6)/2.5)
        B=Z1(3,J)*4.0*(EGO-XORG(1)-XORG(2)-XORG(4))+2.0*Z1(5,J)*XORG(1)
        1+2.0*Z1(7,J)*XORG(1)
        C1=(A1/B)
        C2=EXP((-B)*A1)
        X(5,J+1)=C1*(1.0-C2)+XORG(5)*C2
34982   CONTINUE
        DO 107 I=1,7
        XORG(I)=X(I,J+1)
107     CONTINUE
        CALL FLASH(DIST,NOX,T,J)
        IF(NOX.EQ.0) GOTO 100
        X(5,J+1)=XI(5)
        X(6,J+1)=XI(6)
        X(7,J+1)=XI(7)
        DISTF=DISTF+DIST
100     CONTINUE
        A=A*5.0
        RETURN
        END
C       ********************************************************
C       THIS SUBROUTINE CONTAINS THE STATE VARIABLE EQUATIONS
        SUBROUTINE EQTN1(J)
```

```fortran
      DIMENSION FP(7),XR(7)
      DIMENSION Z1(9,2001)
C------------------------------------------------------------------
      COMMON/A5/Z1/RUNG11/FP,XR
      COMMON/A9/EGO
C------------------------------------------------------------------
      FP(1)=-2.0*Z1(3,J)*(XR(1)*XR(1)-((2.0*XR(5))/0.5)*(1.0-XR(1)-XR
     1(2)-XR(4)))-Z1(4,J)*XR(1)-2.0*Z1(5,J)*XR(1)*XR(5)-2.0*Z1(6,J)*
     1XR(1)*XR(1)+Z1(7,J)*(2.0*XR(2)*XR(5)-(XR(1)*XR(6)/2.5))
     1-Z1(8,J)*(XR(2)*XR(1)-(XR(6)*(EGO-XR(1)-XR(2)-XR(4))/1.25))
     1-Z1(3,J)*XR(4)*XR(1)
      FP(2)=Z1(4,J)*XR(1)+Z1(5,J)*2.0*XR(1)*XR(5)+Z1(6,J)*XR(1)*XR
     1(1)-Z1(7,J)*((2.0*XR(2)*XR(5))-(XR(1)*XR(6)/2.5))
     1-Z1(8,J)*(XR(2)*XR(1)-((XR(6)/1.25)*(EGO-XR(1)-XR(2)-XR(4))))
     1+Z1(9,J)*(FGO-XR(1)-XR(2)-XR(4))/2.0
      FP(3)=2.0*Z1(5,J)*XR(1)*XR(5)+Z1(6,J)*XR(1)*XR(1)
      FP(4)=Z1(9,J)*(EGO-XR(1)-XR(2)-XR(4))/2.0-Z1(3,J)*XR(4)*XR(1)
      FP(5)=Z1(3,J)*(XR(1)*XR(1)-4.0*XR(5)*(EGO-XR(1)-XR(2)-XR(4)))-Z1
     1(7,J)*(2.0*XR(2)*XR(5)-(XR(1)*XR(6)/2.5))-2.0*Z1(5,J)*XR(1)*XR(
     15)
      FP(6)=Z1(7,J)*(2.0*XR(2)*XR(5)-(XR(1)*XR(6)/2.5))+Z1(8,J)*(XR(2)
     1*XR(1)-(XR(6)/1.25)*(EGO-XR(1)-XR(2)-XR(4)))
      FP(7)=Z1(4,J)*XR(1)+Z1(3,J)*XR(1)*XR(4)
      RETURN
      END
C
C     *************************************************************
C
C------------------------------------------------------------------
C     THIS SUBROUTINE IS USED FOR SOLVING THE ADJOINT VARIABLE
C     EQUATIONS USING FOURTH-ORDER RUNGE-KUTTA METHOD
      SUBROUTINE RUNG2(ALPHA2,ALPHA3,A)
      DIMENSION X(7,2001),V(4,2001),Z(9,2001),Z1(9,2001),ZI(9)
      DIMENSION VORG(4),VORG1(4),FP(4),VR(4),EO(9),ZO(9)
      DIMENSION CP(4),CP1(4),CP2(4),CP3(4),CP4(4)
      DIMENSION VW(2001),VDMT(2001),VG(2001),VT(2001)
      DIMENSION VAP(7),TN(2001),TNEW(101),XB(7)
C------------------------------------------------------------------
      COMMON/A2/X/A3/Z
      COMMON/A4/VW,VDMT,VG,VT
      COMMON/A5/Z1/A6/TN/A7/V
      COMMON/A30/TNEW/FF1/PT,H
      COMMON/A9/EGO
      COMMON/RUNG21/FP,VR/E1/EO,ZO,R
C
      REAL MP
      REAL MPP
      A=A/5.0
      DO 200 JI=1,2000
      J1=2002-JI
      J=J1
      MP=(X(1,J)+X(2,J)+X(3,J)+X(4,J))/2.0
C     VT(J)=VDMT(J)*MP+VG(J)*X(5,J)+VW(J)*X(6,J)
      DO 55 I=1,4
      VORG1(I)=V(I,J)
55    CONTINUE
      N=1
      D=0.0
      DO 12 I=3,9
      ZI(I)=Z(I,J)
12    CONTINUE
      DO 399 I=1,7
      XB(I)=X(I,J)
```

```
399        CONTINUE
5555       CONTINUE
           T=TN(J)-D*(TN(J)-TN(J-1))
           DO 1111 I=3,9
           Z(I,J)=60.0*EXP(-EO(I)/(R*T)+ZO(I))
1111       CONTINUE
           DO 398 I=1,7
           X(I,J)=X(I,J)-D*(X(I,J)-X(I,J-1))
398        CONTINUE
           X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
           X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
           X3=(19.422+0.025*(T-413.0))/1000.0
           MPP=(X(1,J)+X(2,J)+X(3,J)+X(4,J))/2.0
           VTO=X1*MPP+X2*X(5,J)+X3*X(6,J)
           DO 8888 I=3,9
           IF((I.EQ.4).OR.(I.EQ.9))GOTO 8888
           Z(I,J)=Z(I,J)/VTO
8888       CONTINUE
           D=D+0.2
           DO 201 N=1,4
           VORG(N)=V(N,J)
           VR(N)=V(N,J)
201        CONTINUE
           CALL EQTN2(ALPHA2,ALPHA3,J,T)
           DO 202 N=1,4
           CP1(N)=A*FP(N)
           VR(N)=VORG(N)-0.5*CP1(N)
202        CONTINUE
           CALL EQTN2(ALPHA2,ALPHA3,J,T)
           DO 203 N=1,4
           CP2(N)=A*FP(N)
           VR(N)=VORG(N)-0.5*CP2(N)
203        CONTINUE
           CALL EQTN2(ALPHA2,ALPHA3,J,T)
           DO 206 N=1,4
           CP3(N)=A*FP(N)
           VR(N)=VORG(N)-CP3(N)
206        CONTINUE
           CALL EQTN2(ALPHA2,ALPHA3,J,T)
           DO 205 N=1,4
           CP4(N)=A*FP(N)
           CP(N)=(CP1(N)+2.0*(CP2(N)+CP3(N))+CP4(N))/6.0
           V(N,J)=VORG(N)-CP(N)
205        CONTINUE
           N=N+1
           IF(N.LE.5)GOTO 5555
           DO 56 N=1,4
           V(N,J-1)=V(N,J)
           V(N,J)=VORG1(N)
56         CONTINUE
           DO 578 I=3,9
           Z(I,J)=Z1(I)
578        CONTINUE
           DO 397 I=1,7
           X(I,J)=XB(I)
397        CONTINUE
200        CONTINUE
           A=A*5.0
           RETURN
           END
C_____
```

```
C         SUBROUTINE TO SOLVE ADJOINT EQUATIONS
          SUBROUTINE EQTN2(ALPHA2,ALPHA3,J,T)
          DIMENSION X(7,2001),Z(9,2001),FP(4),VR(4)
          DIMENSION TN(2001),TNEW(101),VAP(7)
          COMMON/A2/X/A3/Z
          COMMON/A9/EGU
          COMMON/RUNG21/FP,VR
          COMMON/A6/TN/A30/TNEW/FF1/PT,H
C         CALCULATION OF THE FIRST DERIVATIVES OF "F" W.R.T."X"
          T=TN(J)
          VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
          VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
          P=PT/(2.0*(VAP(6)-PT))
          WX1=P;WX2=P;WX3=P;WX4=P
          Q=PT/(2.0*(VAP(5)-PT))
          GX1=Q;GX2=Q;GX3=Q;GX4=Q
          DF1X1=-4.0*Z(3,J)*X(1,J)+((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGU-X(1,J)
         1-X(2,J)-X(4,J))*GX1)-Z(4,J)-((2.0*Z(5,J))*(X(5,J)+X(1,J)*GX1
         1))-4.0*Z(6,J)*X(1,J)+2.0*Z(7,J)*X(2,J)*GX1-((Z(7,J)/2.5)*(X(1,
         1J)*WX1+X(6,J)))-Z(8,J)*X(2,J)+((Z(8,J)/1.25)*(-X(6,J)+(EGU-X(
         11,J)-X(2,J)-X(4,J))*WX1))-Z(3,J)*X(4,J)
          DF1X2=((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGU-X(1,J)-X(2,J)-X(4,J))*GX2
         1))-2.0*Z(5,J)*X(1,J)*GX2+((2.0*Z(7,J))*(X(5,J)+X(2,J)*GX2))-
         1(Z(7,J)*X(1,J)/2.5)*WX2-Z(8,J)*X(1,J)+((Z(8,J)/1.25)*(-X(6,J)
         1+(EGU-X(1,J)-X(2,J)-X(4,J))*WX2))
          DF1X3=0.0
          DF1X4=((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGU-X(1,J)-X(2,J)-X(4,J))*
         1GX4))-2.0*Z(5,J)*X(1,J)*GX4+2.0*Z(7,J)*X(2,J)*GX4-(Z(7,J)*X(1,
         1J)/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,J)+(EGU-X(1,J)-X(2,J)-X(4,J))
         1*WX4))-Z(3,J)*X(1,J)
          DF2X1=Z(4,J)+((2.0*Z(5,J))*(X(1,J)*GX1+X(5,J)))+2.0*Z(6,J)*X(1,J
         1)-2.0*Z(7,J)*X(2,J)*GX1+((Z(7,J)/2.5)*(X(1,J)*WX1+X(6,J)))-Z(8,
         1J)*X(2,J)+((Z(8,J)/1.25)*(-X(6,J)+(EGU-X(1,J)-X(2,J)-X(4,J))*
         1WX1))-Z(9,J)/2.0
          DF2X2=2.0*Z(5,J)*X(1,J)*GX2-((2.0*Z(7,J))*(X(2,J)*GX2+X(5,J)))+
         1(Z(7,J)*X(1,J)/2.5)*WX2-Z(8,J)*X(1,J)+((Z(8,J)/1.25)*(-X(6,J)+
         1(EGU-X(1,J)-X(2,J)-X(4,J))*WX2))-Z(9,J)/2.0
          DF2X3=0.0
          DF2X4=2.0*Z(5,J)*X(1,J)*GX4-2.0*Z(7,J)*X(2,J)*GX4+(Z(7,J)*X(1,J)
         1/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,J)+(EGU-X(1,J)-X(2,J)-X(4,J))*
         1WX4))-Z(9,J)/2.0
          DF3X1=((2.0*Z(5,J))*(X(5,J)+X(1,J)*GX1))+2.0*Z(6,J)*X(1,J)
          DF3X2=2.0*Z(5,J)*X(1,J)*GX2
          DF3X3=0.0
          DF3X4=2.0*Z(5,J)*X(1,J)*GX4
          DF4X1=-Z(9,J)/2.0-Z(3,J)*X(4,J)
          DF4X2=-Z(9,J)/2.0
          DF4X3=0.0
          DF4X4=-Z(9,J)/2.0-Z(3,J)*X(1,J)
C
C         --------------------------------------------------------------
C         CALCULATION OF "FP"
          FP(1)=(2.0*ALPHA2*EGU*((EGU/X(1,J))-15.0))/(225.0*X(1,J)*X(1,J))
         1-(DF1X1*VR(1)+DF2X1*VR(2)+DF3X1*VR(3)+DF4X1*VR(4))
          FP(2)=-2.0*ALPHA3*X(2,J)-(DF1X2*VR(1)+DF2X2*VR(2)+DF3X2*VR(3)
         1+DF4X2*VR(4))
          FP(3)=-(DF1X3*VR(1)+DF2X3*VR(2)+DF3X3*VR(3)+DF4X3*VR(4))
          FP(4)=-2.0*ALPHA3*X(4,J)-(DF1X4*VR(1)+DF2X4*VR(2)+DF3X4*VR(3)
         1+DF4X4*VR(4))
          RETURN
          END
C         --------------------------------------------------------------
```

```
C     *******************************************************************
C     THIS SUBROUTINE CALCULATES THE DERIVATIVE OF THE HAMILT-
C     ONIAN W.R.T. TEMP
      SUBROUTINE DHU(M)
      DIMENSION HP(101),X(7,2001),ZP(9,101),V(4,2001),Z(9,2001)
      DIMENSION VAP(7),TNEW(101),VT(2001),VW(2001),VG(2001),VDMT(2001)
      COMMON/A8/ZP,HP/A2/X/A7/V
      COMMON/A9/EGO/FF1/PT,H/A30/TNEW/A3/Z/A4/VW,VDMT,VG,VT
      DO 30 J1=1,M
      J=J1
      T=TNEW(J)
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      A=VAP(5)*((3729.0/(T*T))-4.042/T)
      B=VAP(6)*(1757.853/(T-33.274)**2)
      C=-(X(5,J)*A+X(6,J)*B)
      GT=C/(VAP(5)-PT)
      WT=C/(VAP(6)-PT)
      HP(J)=0.0
      HP(J)=V(1,J)*(-2.0*ZP(3,J)*(X(1,J)*X(1,J)-((4.0*X(5,J)/0.50)*(1.
     10-X(1,J)-X(2,J)-X(4,J)))-ZP(4,J)*X(1,J)-2.0*ZP(5,J)*X(1,J)*X(5
     1,J)-2.0*ZP(6,J)*X(1,J)*X(1,J)+ZP(7,J)*(2.0*X(2,J)*X(5,J)-(X(1,J
     1)*X(6,J)/2.5))-ZP(8,J)*(X(2,J)*X(1,J)-(X(6,J)/1.25)*(1.0-X(1,J)
     1-X(2,J)-X(4,J)))-ZP(3,J)*X(4,J)*X(1,J))+HP(J)
      HP(J)=HP(J)+V(1,J)*((4.0*Z(3,J+1)*(1.0-X(1,J)-X(2,J)-X(4,J))
     1*GT/0.5)-2.0*X(1,J)*Z(5,J+1)*GT+2.0*X(2
     1,J)*Z(7,J+1)*GT-(Z(7,J+1)*X(1,J)*WT/2.5)
     1+(Z(8,J+1)*(1.0-X(1,J)-X(2,J)-X(4,J))*WT/1.25))
      HP(J)=HP(J)+V(2,J)*(ZP(4,J)*X(1,J)+2.0*ZP(5,J)*X(1,J)*X(5,J)+
     1ZP(6,J)*X(1,J)*X(1,J)-ZP(7,J)*(2.0*X(2,J)*X(5,J)-(X(1,J)*X(6,J
     1)/2.5))-ZP(8,J)*(X(2,J)*X(1,J)-(X(6,J)/1.25)*(1.0-X(2,J)-X(1,J)
     1-X(4,J)))+ZP(9,J)*(EGO-X(1,J)-X(2,J)-X(4,J))/2.0)
      HP(J)=HP(J)+V(2,J)*(2.0*X(1,J)*Z(5,J+1)*GT-2.0*X(2,J)*
     1Z(7,J+1)*GT+(Z(7,J+1)*X(1,J)*WT/2.5)+(Z(8,J+1)*
     1(1.0-X(1,J)-X(2,J)-X(4,J))/1.25)*WT)
      HP(J)=HP(J)+V(3,J)*(2.0*ZP(5,J)*X(1,J)*X(5,J)+ZP(6,J)*X(1,J)
     1*X(1,J))
      HP(J)=HP(J)+V(3,J)*(2.0*X(1,J)*Z(5,J+1)*GT)
      HP(J)=HP(J)+V(4,J)*((ZP(9,J)/2.0)*(EGO-X(1,J)-X(2,J)-X(4,J))
     1-ZP(3,J)*X(4,J)*X(1,J))
30    CONTINUE
      RETURN
      END
C     *******************************************************************
C     THIS SUBROUTINE INTEGRATES THE INTEGRAL TERM IN THE OBJECTI-
C     VE FUNCTION USING SIMPSONS RULE
      SUBROUTINE SIMPSN
      DIMENSION X(7,2001),SX(6,2001)
      COMMON/A1/SSX(6)/A2/X
      COMMON/A9/EGO
      DO 1008 K=1,4
      SSX(K)=0.0
1008  CONTINUE
      DO 666 I=2,4,2
      DO 99 J=3,1999,2
      SX(I,J)=SSX(I)+2.0*(X(I,J)*X(I,J))
      SSX(I)=SX(I,J)
99    CONTINUE
      DO 108 J=2,2000,2
      SX(I,J)=SSX(I)+4.0*(X(I,J)*X(I,J))
      SSX(I)=SX(I,J)
```

```
108     CONTINUE
        SSX(I)=(1/6003.0)*(SSX(I)+(X(I,1)**2)+(X(I,2001)**2))
666     CONTINUE
        DO 5000 J=3,1999,2
        SX(1,J)=SSX(1)+2.0*((EGO/X(1,J))-15.0)*((EGO/X(1,J))-15.0)
        SSX(1)=SX(1,J)
5000    CONTINUE
        DO 5001 J=2,2000,2
        SX(1,J)=SSX(1)+4.0*((EGO/X(1,J))-15.0)*((EGO/X(1,J))-15.0)
        SSX(1)=SX(1,J)
5001    CONTINUE
        SSX(1)=(1.0/6003.0)*(SSX(1)+(((EGO/X(1,1))-15.0)**2)+
       1(((EGO/X(1,2001))-15.0)**2))
        RETURN
        END
C       ***********************************************************
C       THIS SUBROUTINE CALCULATES THE AMOUNT OF VOLATILE COMP-
C       ONENTS(ETHYLENE GLYCOL,WATER,ACETALDEHYDE)FLASHED FROM
C       THE REACTION MASS
        SUBROUTINE FLASH(DIST,NOX,T,J)
        DIMENSION XORG(7),XI(7),F(7),AK(7),FNEW(7)
        DIMENSION VW(2001),VDMT(2001),VG(2001),VT(2001)
C_____
        COMMON/R1/XI,XORG
        COMMON/F1/AK,F
        COMMON/FF1/PT,H
        COMMON/A4/VW,VDMT,VG,VT
C_____
        FTOT=(XORG(1)+XORG(2)+XORG(3)+XORG(4))/2.0
        DO 701 I=5,7
        FTOT=FTOT+XORG(I)
701     CONTINUE
        DO 702 I=1,7
        F(I)=XORG(I)/FTOT
        AK(I)=0.0
702     CONTINUE
        AK(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
        AK(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
        AK(7)=(55.0*EXP(41.215329-(4860.8854/T)-(5.04843*ALOG(T))+
       1(3.05143E-17)*(T**6.0)))
        PT1=((F(5)*AK(5)+F(6)*AK(6)+F(7)*AK(7)))
        IF(PT1.LT.PT) GOTO 703
707     CALL CONV(ROOT,NOX,J)
        IF(NOX.GT.10) GOTO 710
        DO 704 I=1,7
        FNEW(I)=F(I)/(1.0+ROOT*((AK(I)*EXP(1.0+H)/PT)-1.0))
704     CONTINUE

        FNEWT=(1.0-ROOT)*FTOT
        DIST=FTOT*ROOT
        DO 705 I=1,7
        XI(I)=(FNEW(I)*FNEWT)
        IF(XI(I).GE.0.0) GOTO 705
705     CONTINUE
119     FORMAT(//)
        NOX=1
        IF((NOX-MOX).EQ.0) GOTO 706
        M=0
        GOTO 706
703     NOX=0
        GOTO 706
```

```
708        FORMAT(1X,'** VAC. APPLIED AT J=',I5,'**')
710        TYPE 711,J
711        FORMAT(1X,'* SOMETHING HAS GONE WRONG',/,'CHECK! J=',I4)
706        IF(NOX.EQ.0) GOTO 709
709        RETURN
           END
C          ***********************************************************************
C          THIS SUBROUTINE FINDS THE ROOT USING NEWTONS METHOD
           SUBROUTINE CONV(ROOT,NOX,J)
           DIMENSION AK(7),F(7)
C_____
           COMMON/F1/AK,F
           COMMON/FF1/PT,H
C_____
           R=0.00001
           ITN=0
35         FUNR=0.0
           DFUNR=0.0
           DO 1100 I=5,7
           FUNR=FUNR+(AK(I)*F(I))/(PT*(1.0+R*((AK(I)/PT)-1.0)))
           DFUNR=DFUNR+(-AK(I)*F(I)*((AK(I)/PT)-1.0))/(PT*((1.0
          1+R*((AK(I)/PT)-1.0))**2))
1100       CONTINUE
           FUNR=FUNR-1.0
           IF(ABS(FUNR).LE.1.0E-06) GO TO 80
           IF(ITN.GT.20) GO TO 60
           R=R-FUNR/DFUNR
           ITN=ITN+1
           GO TO 35
60         TYPE 40
40         FORMAT(15X,'ITERATION DOES NOT CONVERGE')
80         IF(R.GT.0.0)ROOT=R
           CONTINUE
           NOX=1
           RETURN
           END
C          ***********************************************************************
```

2. COMPUTER PROGRAM FOR THE SECOND VARIATION METHOD

```
************************************************************************
*      OPTIMIZATION OF PET REACTORS                              *
************************************************************************

       THIS PROGRAM USES THE SECOND VARIATION METHOD FOR FINDING
       THE OPTIMAL TEMPERATURE PROFILES IN A BATCH PET REACTOR.

       THIS METHOD IS CARRIED ON TILL THE OBJECTIVE FUNCTION
       DOES NOT CHANGE FROM ITERATION TO ITERATION

       NEAR OPTIMAL TEMPERATURE PROFILE OBTAINED FROM THE CONTROL
       VECTOR ITERATION METHOD IS USED AS THE INITIAL GUESS FOR
       THIS METHOD

************************************************************************

                   MAIN PROGRAM

************************************************************************
       X          STATE VARIABLES
       V          ADJOINT VARIABLES
       E0         ACTIVATION ENERGY
       Z0         FREQUENCY FACTOR
       Z          RATE CONSTANTS FOR THE REACTIONS INVOLVED IN THE POLY-
                  CONDENSATION STAGE OF PET FORMATION.
       A          WIDTH OF THE TIME INTERVAL USED IN INTEGRATIONS WITH A
                  TOTAL TIME OF 2 HRS.
       EG0        INITIAL CONCENTRATION OF EG GROUP IN THE FEED
       PT         TOTAL PRESSURE,ATM.
       T          REACTOR TEMPERATURE,DEGREE KELVIN.
       TNEW       TEMPERATURE AT 101 POINTS OF TIME
       TN         TEMPERATURE AT 2001 POINTS OF TIME
       R          UNIVERSAL GAS CONSTANT
       ZP         DERIVATIVE OF Z W.R.T. T
       VDMT       MOLAR VOLUME OF DMT
       VG         MOLAR VOLUME OF GLYCOL
       VW         MOLAR VOLUME OF WATER
       VT         TOTAL VOLUME OF THE REACTION MASS,LITRES
       Z1         Z DIVIDED BY VT
       PI         OBJECTIVE FUNCTION
       VP         DERIVATIVE OF VT W.R.T. T
       EPSL       EPSILON-FACTOR DETERMINING THE MAGNITUDE OF STEP TO BE
                  TAKEN IN THE CORRECTION OF TEMPERATURE FROM THE PREVI-
                  OUS ITERATON
       EOPT       OPTIMUM EPSILON
       AK         VAPOUR PRESSURE,ATM
                  ALPHA1 - WEIGHTING PARAMETER FOR THE DESIRED CONCENTR-
                  ATION OF DIETHYLENE GLYCOL IN THE POLYMER
                  ALPHA2 - WEIGHTING PARAMETER FOR THE DESIRED VALUE OF
                  DPBAR OF THE POLYMER
                  ALPHA3 - WEIGHTING PARAMETER FOR THE FORMATION OF
                  SIDE PRODUCTS IN THE REACTON MASS
       DPBAR      NUMBER-AVERAGE MOLECULAR WEIGHT OF THE POLYMER
       P          VARIABLE INVOLVED IN THE RICCATI EQUATION.THIS IS
                  A (4X4) MATRIX
       Q          VARIABLE INVOLVED IN THE Q EQUATIONS.THIS IS A (4X1)
                  VECTOR FUNCTION OF TIME
       DELLX      PERTURBATION IN THE STATE VARIABLES
```

```
*        HT1        FIRST DERIVATIVE OF THE HAMILTONIAN W.R.T. TEMP
*        HT2        SECOND DERIVATIVE OF THE HAMILTONIAN W.R.T. TEMP
*        RG         UNIVERSAL GAS CONSTANT
*
****************************************************************
*
         DIMENSION  X(7,2001),V(4,2001),ZP(9,101)
         DIMENSION  T(20),TNEW(101),TR(101),TU(2001)
         DIMENSION  Z(9,2001),Z1(9,2001),Z0(9),E0(9)
         DIMENSION  VT(2001),VDMT(2001),VN(2001),VG(2001)
         DIMENSION  DPBAR(2001),EP(15),OI(15),E3(15)
         DIMENSION  P(4,1,101),DFTP(1,4,101)
         DIMENSION  O(4,1,101),OR(4,1)
         DIMENSION  DELLX(4,1,101),DFT(1,4),R(1,1,101),PR(1,1)
         DIMENSION  HT2(101),FX(4,4),XF(4,4)
         DIMENSION  HT1(101),DELLXR(4,1),SX(6,2001)
****************************************************************
         COMMON/A1/SSX(6)/A2/X/A3/Z
         COMMON/A4/VN,VDMT,VG,VT/E1/E0,Z0,RG
         COMMON/A5/Z1/A6/TU/A7/V
         COMMON/A9/EGO
         COMMON/FF1/PT,H
         COMMON/B8/DISTF
         COMMON/A12/DPBAR
         COMMON/A13/P/B15/DFTP
         COMMON/A16/R/A18/HT2
         COMMON/A19/FX,XF
         COMMON/A24/O/A25/DELLX
         COMMON/A26/HT1
****************************************************************
         REAL MP
         EGO=1.0
         H=-1.0
         WRITE(40,540)
         TYPE 540
540      FORMAT(1X,'GIVE THE VALUES OF ALPHA1,ALPHA2,ALPHA3 IN
         1DOUBLE LINES')
         READ(5,*),ALPHA1,ALPHA2,ALPHA3
         WRITE(40,3),ALPHA1,ALPHA2,ALPHA3
3        FORMAT(1X,'ALPHA1=',F15.5,2X,'ALPHA2=',F15.5,2X,'ALPHA3=',
         1F15.5)
         M=101
         WRITE(40,673)
         TYPE 673
673      FORMAT(1X,'GIVE TOTAL PRESSURE')
         READ(5,*),PT
         WRITE(40,573),PT
573      FORMAT(1X,'PT=',F15.5)
         OPEN(UNIT=23)
         READ(23,549),(TNEW(J),J=1,M)
549      FORMAT(1X,101F7.2)
1213     FORMAT(1X,'GUESS TEMP PROFILE FOR THE SECOND ORDER METHOD (S')
         WRITE(40,1213)
         WRITE(40,549),(TNEW(J),J=1,101)
         A=1.0/1000.0
         RG=1.987
         NO=0
         E0(3)=18500.00;E0(4)=29800.00
         E0(5)=29800.00;E0(6)=29800.00
         E0(7)=17600.00;E0(8)=17600.00
```

```fortran
      EO(9)=37600.00
      ZO(3)=13.43;ZO(4)=17.54;ZO(5)=17.54
      ZO(6)=17.54;ZO(7)=13.855;ZO(8)=13.855
      ZO(9)=22.004
11    NO=NO+1
      WRITE(40,12),NO
      TYPE 12,NO
12    FORMAT(1X,'NO=',I2)
      WRITE(40,16)
      WRITE(40,16)
16    FORMAT(1X,120(1H_))
C     THIS PART CALCULATES THE MOLAR VOLUMES AND RATE CONSTANTS
C     AT 2001 POINTS OF TIME
      DO 19 I=3,9
      Z(I,1)=60.0*EXP(-EO(I)/(RG*TNEW(1))+ZO(I))
      ZP(I,1)=EO(I)*Z(I,1)/(RG*TNEW(1)*TNEW(1))
19    CONTINUE
      TN(1)=TNEW(1)
      VDMT(1)=191.5*(1.0+0.0014*(TN(1)-413.0))/1000.0
      VG(1)=60.6*(1.0+0.0014*(TN(1)-413.0))/1000.0
      VW(1)=(19.422+0.025*(TN(1)-413.0))/1000.0
      J1=2
      DO 14 J=1,100
      D=1.0
      DO 14 I=1,20
      T(I)=(TNEW(J+1)-TNEW(J))*D/20.0+TNEW(J)
      DO 15 I1=3,9
      Z(I1,J1)=60.0*EXP(-EO(I1)/(RG*T(I))+ZO(I1))
15    CONTINUE
      TN(J1)=T(I)
      VDMT(J1)=191.5*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VG(J1)=60.6*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VW(J1)=(19.422+0.025*(TN(J1)-413.0))/1000.0
      D=D+1.0
      J1=J1+1
14    CONTINUE
C     INITIAL CONDITIONS ON THE STATE VARIABLES
      X(1,1)=1.0
      X(2,1)=0.0
      X(3,1)=0.0
      X(4,1)=0.0
      X(5,1)=0.0
      X(6,1)=0.0
      X(7,1)=0.0
      CALL RUNG1(A)
678   FORMAT(2X,'X IS:',21E10.3)
      WRITE(40,678),((X(I,J),J=1,2001,100),I=1,7)
      WRITE(40,16)
      TYPE 791,NO,DISTF
791   FORMAT(1X,'FOR NO=',I2,'AMOUNT FLASHED=',F15.8)
      WRITE(40,791),NO,DISTF
      DISTF=0.0
C     FINAL CONDITIONS ON THE ADJOINT VARIABLES
      V(1,2001)=0.0
      V(2,2001)=0.0
      V(3,2001)=ALPHA1
      V(4,2001)=0.0
      MP=(X(1,2001)+X(2,2001)+X(3,2001)+X(4,2001))/2.0
      VT(2001)=VDMT(2001)*MP+VG(2001)*X(5,2001)+VW(2001)*X(6,2001)
      Z1(4,2001)=Z(4,2001)
      Z1(9,2001)=Z(9,2001)
```

```
      DO 191 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 191
      Z1(I,2001)=(Z(I,2001)/VT(2001))
191   CONTINUE
      WRITE(40,16)
      CALL RUNG2(ALPHA2,ALPHA3,A)
      WRITE(40,600),((V(I,J),J=1,2001,100),I=1,4)
      TYPE 12,NO
600   FORMAT(1X,'V IS:',21E11.3)
      CALL SIMPSN
      PI=ALPHA1*X(3,2001)+(ALPHA2/225.0)*SSX(1)+ALPHA3*(SSX(2)
     1+SSX(4))
      TYPE 26,PI
26    FORMAT(1X,'PI=',F20.7)
      WRITE(40,16)
      WRITE(40,28),PI
      TYPE 113
      READ(5,455),C1
      IF(C1.EQ.'N')GOTO 42
      IF(NO.GT.5)GOTO 42
C     FINAL CONDITIONS ON THE "P" DIFFERENTIAL EQUATIONS
      DO 29 I=1,4
      DO 29 K=1,4
      P(I,K,101)=0.0
29    CONTINUE
C     FINAL CONDITIONS ON THE "Q" DIFFERENTIAL EQUATIONS
      DO 79 K=1,4
      I=1
      Q(K,I,101)=0.0
79    CONTINUE
C     INITIAL CONDITIONS ON THE PERTURBATION EQUATIONS OF THE
C     STATE VARIABLES
      DO 89 K=1,4
      I=1
      DELLX(K,I,1)=0.0
89    CONTINUE
      CALL RUNG3(ALPHA2,ALPHA3,A)
      TYPE 1600,(((P(I,K,J),J=1,101,10),K=1,4),I=1,4)
      WRITE(40,16)

      CALL RUNG4(ALPHA2,ALPHA3,A)
      TYPE 1700,((Q(1,1,J),J=1,101,10),I=1,4)
      TYPE 113
      READ(5,455),E1
      IF(E1.EQ.'Y')GOTO 13000
      GOTO 13001
13000 CALL RUNG5(ALPHA2,ALPHA3,A)
      TYPE 1800,((DELLX(I,1,J),J=1,101,10),I=1,4)
1600  FORMAT(1X,'P IS:',11E11.3)
1700  FORMAT(1X,'Q IS:',11E11.3)
1800  FORMAT(1X,'DELLX IS:',11E11.3)
13001 WRITE(60,12),NO
C**********************************************************
C     THIS PART CALCULATES THE OPTIMUM EPSILON
121   TYPE 2929
      DO 601 K2=1,10
      EP(K2)=0.0
      QI(K2)=0.0
601   CONTINUE
      K1=1
2929  FORMAT(1X,'GIVE THE VALUE OF STARTING EPSILON')
```

```
      READ(5,*),EPSL
      E3(NO)=EPSL
      WRITE(60,7003),E3(NO)
7003  FORMAT(1X,'THE STARTING EPSILON=',F15.5)
      WRITE(40,7003),E3(NO)
57    DO 5151 J1=1,101
      J=J1
      K=1
      DO 52 I=1,4
      OR(I,K)=O(I,K,J)
      RR(K,I)=R(K,I,J)
      DELLXR(I,K)=DELLX(I,K,J)
      DFT(K,I)=DFTP(K,I,J)
52    CONTINUE
C     MULTIPLICATION OF THE MATRICES "OR" AND "DFT"
      DOFT=0.0
      DO 21 I=1,4
      DOFT=DOFT+DFT(K,I)*OR(I,K)
21    CONTINUE
C     MULTIPLICATION OF THE MATRICES "RR" AND "DELLXR"
      RDELLX=0.0
      DO 22 I=1,4
      RDELLX=RDELLX+RR(K,I)*DELLXR(I,K)
22    CONTINUE
      TERM1=(HT1(J)+DOFT)*EPSL*HT2(J)
      TERM2=HT2(J)*RDELLX
      TR(J)=TNEW(J)-TERM1-TERM2
      TR(J)=TNEW(J)+TERM1+TERM2
C     IF(TR(J).LT.600.0)GOTO 5
      TR(J)=600.0
5     IF(TR(J).GT.400.0)GOTO 5151
      TR(J)=400.0
5151  CONTINUE
      DO 60 I=3,9
      Z(I,1)=60.0*EXP(-EU(I)/(RG*TR(1))+ZO(I))
60    CONTINUE
      VDMT(1)=191.5*(1.0+0.0014*(TR(1)-413.0))/1000.0
      VG(1)=60.6*(1.0+0.0014*(TR(1)-413.0))/1000.0
      VW(1)=(19.422+0.025*(TR(1)-413.0))/1000.0
      J1=2
      TN(1)=TR(1)
      DO 152 J=1,100
      D=1.0
      DO 152 I=1,20
      T(I)=(TR(J+1)-TR(J))*D/20.0+TR(J)
      DO 53 I1=3,9
      Z(I1,J1)=60.0*EXP(-EU(I1)/(RG*T(I))+ZO(I1))
53    CONTINUE
      TN(J1)=T(I)
      VDMT(J1)=191.5*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VG(J1)=60.6*(1.0+0.0014*(TN(J1)-413.0))/1000.0
      VW(J1)=(19.422+0.025*(TN(J1)-413.0))/1000.0
      D=D+1.0
      J1=J1+1
152   CONTINUE
      CALL RUNG1(A)
      DISTF=0.0
      TYPE 12,NO

      TYPE 166,K1
```

```
166       FORMAT(1X,'NO.INSIDE THE LOOP=',I2)
          TYPE 678,((X(3,J),J=1,2001,100))
          CALL SIMPSN
          PTIP=ALPHA1*X(3,2001)+(ALPHA2/225.0)*SSX(1)+ALPHA3*(SSX(2)
         1+SSX(4))
          TYPE 12,NO
          EP(K1)=EPSL
          TYPE 82
82        FORMAT(1X,'MORE?')
          READ(5,455),B1
          IF(B1.EQ.'N')GOTO 72
          OI(K1)=PINEW
          TYPE 71,EP(K1),OI(K1)
          TYPE 82
          READ(5,455),B1
          IF(B1.EQ.'N')GOTO 72
          EPSL=EPSL/2.0
          K1=K1+1
          IF(K1.LT.11)GOTO 57
C*****************************************************************
72        TYPE 12,NO
          TYPE 123
123       FORMAT(1X,'DO YOU WANT TO GO BACK IN E.OPT.LOOP(Y/N)?')
          READ(5,455),B1
          IF(B1.EQ.'Y')GOTO 121
          WRITE(60,16)
          TYPE 12,NO
          WRITE(40,16)
          WRITE(40,65)
          WRITE(60,65)
          TYPE 65
65        FORMAT(15X,'1',30X,'E')
          DO 70 K1=1,10
          WRITE(40,71),OI(K1),EP(K1)
          WRITE(60,71),OI(K1),EP(K1)
71        FORMAT(7X,F17.7,3X,F17.7)
70        CONTINUE
64        FORMAT(2X,'FOPT=',E20.10)
          TYPE 1008
1008      FORMAT(1X,'GIVE THE VALUE OF FOPT')
          READ(5,*),AL
          FOPT=AL
          WRITE(60,16)
          WRITE(40,16)
          WRITE(40,64),FOPT
          WRITE(60,64),FOPT
C****************************************************************
C         THIS PART CORRECTS THE OLD TEMPERATURE PROFILE USING
C         THE OPTIMUM EPSILON
          DO 5152 J1=1,101
          J=J1
          K=1
          DO 5252 I=1,4
          QR(I,K)=Q(I,K,J)
          RR(K,I)=R(K,I,J)
          DELLXR(I,K)=DELLX(I,K,J)
          DFT(K,I)=DFTP(K,I,J)
5252      CONTINUE
C         MULTIPLICATION OF THE MATRICES "QR" AND "DFT"
          DQFT=0.0
          DO 2121 I=1,4
```

```fortran
      DOFT=DOFT+DFT(K,I)*QR(I,K)
2121  CONTINUE
C     MULTIPLICATION OF THE MATRICES "RR" AND "DELLXR"
      RDELLX=0.0
      DO 2225 I=1,4
      RDELLX=RDELLX+RP(K,I)*DFLLXP(I,K)
2225  CONTINUE
      TERM1=(H11(J)+DOFT)*EOPT*HT2(J)
      TERM2=HT2(J)*RDELLX
      TP(J)=TNEW(J)-TERM1-TERM2
C     TP(J)=TNEW(J)+TERM1+TERM2
      IF(TR(J).LT.600.0)GOTO 7
      TP(J)=600.0
7     IF(TR(J).GT.400.0)GOTO 5152
      TR(J)=400.0
5152  CONTINUE
C*****************************************************************
      DO 929 J=1,101
      TNEW(J)=TP(J)
929   CONTINUE
      WRITE(40,16)
      WRITE(60,16)
      TYPE 129,P1,PINEW
129   FORMAT(1X,'P1=',F15.8,5X,'PINEW=',F15.8)
      TYPE 113
      READ(5,455),H1
      IF(H1.EO.'Y')GOTO 1158
      GOTO 42
1158  IF((ABS(PI-PINEW)).LE.0.0001)GOTO 42
      IF(JD.GT.5)GOTO 42
      TYPE 45,(TNEW(J),J=1,101)
      WRITE(40,45),(TNEW(J),J=1,M)
      WRITE(40,16)
455   FORMAT(A1)
113   FORMAT(1X,'DO YOU WANT TO PROCEED(Y/N)!')
      GOTO 11
42    TYPE 45,(TNEW(J),J=1,101)
      WRITE(60,45),(TNEW(J),J=1,101)
      WRITE(52,549),(TNEW(J),J=1,M)
      DO 46 J=1,101
      IF(TNEW(J).LT.600.0)GOTO 4
      TNEW(J)=600.0
4     IF(TNEW(J).GT.400.0)GOTO 46
      TNEW(J)=400.0
46    CONTINUE
45    FORMAT(1X,'TNEW IS:',101F7.2)
      WRITE(40,16)
      DPBAR(2001)=(EGO/(X(1,2001)+X(2,2001)+X(3,2001)+X(4,2001)))
      WRITE(50,899),(X(1,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(2,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(3,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(4,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(5,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(6,J),J=1,2001,20)
      WRITE(50,16)
      WRITE(50,899),(X(7,J),J=1,2001,20)
```

```fortran
      WRITE(50,16)
      WRITE(50,898),(DPBAR(I),I=1,2001,20)
      WRITE(50,16)
899   FORMAT(1X,'X IS:',101F15.6)
898   FORMAT(1X,'DPBAR IS:',101F15.6)
      WRITE(40,12),NO
      STOP
      END
C-----------------------------------------------------------
C     THIS SUBROUTINE INTEGRATES THE STATE VARIABLE EQUATIONS
C     USING THE FOURTH-ORDER RUNGE-KUTTA METHOD
      SUBROUTINE RUNG1(A)
      DIMENSION X(7,2001),XORG(7),CP(7),CP1(7),CP2(7),TN(2001),XI(7)
      DIMENSION CP3(7),CP6(7),FP(7),Z1(9,2001),VT(2001),VC(2001)
      DIMENSION XP(7),Z(9,2001),VDMT(2001),VW(2001),DPBAR(2001)
      DIMENSION XORG1(7),ZI(9),EO(9),ZO(9)
      COMMON/A2/X/A4/VW,VDMT,VG,VT/A3/Z/A5/Z1/A6/TN
      COMMON/RUNG11/FP,XR/E1/EO,ZO,RG
      COMMON/R1/XI,XORG
      COMMON/A9/EGO
      COMMON/FE1/PT,M
      COMMON/B6/DISTF
      COMMON/A12/DPBAR
      REAL MP
      REAL MPP
      A=A/5.0
      DO 100 J1=1,2000
      J=J1
      MP=(X(1,J)+X(2,J)+X(3,J)+X(4,J))/2.0
      DPBAR(J)=((EGO/MP)/2.0)
      VT(J)=VDMT(J)*MP+VG(J)*X(5,J)+VW(J)*X(6,J)
131   DO 99 N=1,7
      XORG1(N)=X(N,J)
99    CONTINUE
      M=1
      D=0.0
      DO 12 I=3,9
      ZI(I)=Z(I,J)
12    CONTINUE
4444  CONTINUE
      DO 101 N=1,7
      XORG(N)=X(N,J)
      XR(N)=X(N,J)
101   CONTINUE
      T1=TN(J)+D*(TN(J+1)-TN(J))
      X1=191.5*(1.0+0.0014*(T1-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T1-413.0))/1000.0
      X3=(19.422+0.025*(T1-413.0))/1000.0
      MPP=(XORG(1)+XORG(2)+XORG(3)+XORG(4))/2.0
      VTO=X1*MPP+X2*XORG(5)+X3*XORG(6)
      DO 11111 I=3,9
      Z(I,J)=60.0*EXP(-EO(I)/(RG*T1)+ZO(I))
11111 CONTINUE
      DO 22222 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 22222
      Z1(I,J)=Z(I,J)/VTO
22222 CONTINUE
      Z1(4,J)=Z(4,J)
      Z1(9,J)=Z(9,J)
      D=D+0.2
      CALL EQTN1(J)
```

```
      DO 102 I=1,7
      CP1(I)=A*FP(I)
      XR(I)=XORG(I)+0.5*CP1(I)
102   CONTINUE
      CALL EQTN1(J)
      DO 103 I=1,7
      CP2(I)=A*FP(I)
      XR(I)=XORG(I)+0.5*CP2(I)
103   CONTINUE
      CALL EQTN1(J)
      DO 106 I=1,7
      CP3(I)=A*FP(I)
      XR(I)=XORG(I)+CP3(I)
106   CONTINUE
      CALL EQTN1(J)
      DO 105 I=1,7
      CP6(I)=A*FP(I)
      CP(I)=(CP1(I)+2.0*(CP2(I)+CP3(I))+CP6(I))/6.0
      X(I,J)=XORG(I)+CP(I)
105   CONTINUE
      J=J+1
      IF(J.LE.5) GOTO 4441
      DO 999 I=1,7
      X(I,J+1)=X(I,J)
      X(I,J)=XORG1(I)
999   CONTINUE
      DO 578 I=3,9
      Z(I,J)=ZI(I)
578   CONTINUE
      T=TN(J)
      IF(X(5,J+1).GT.0.0) GOTO 34982
      A1=Z1(3,J)*XORG(1)*XORG(1)+(Z1(7,J)*XORG(1)*XORG(6)/2.5)
      B=Z1(3,J)*4.0*(EGO-XORG(1)-XORG(2)-XORG(4))+2.0*Z1(5,J)*XORG(1)
     1+2.0*Z1(7,J)*XORG(1)
      C1=(A1/B)
      C2=EXP((-B)*A1)
      X(5,J+1)=C1*(1.0-C2)+XORG(5)*C2
34982 CONTINUE
      DO 107 I=1,7
      XORG(I)=X(I,J+1)
107   CONTINUE
      CALL FLASH(DIST,NOX,T,J)
      IF(NOX.EQ.0) GOTO 100
      X(5,J+1)=XI(5)
      X(6,J+1)=XI(6)
      X(7,J+1)=XI(7)
      DISTF=DISTF+DIST
100   CONTINUE
      A=A*5.0
      RETURN
      END
C     ***********************************************************
C     THIS SUBROUTINE CONTAINS THE STATE VARIABLE EQUATIONS
      SUBROUTINE EQTN1(J)
      DIMENSION FP(7),XR(7),Z1(9,2001)
      COMMON/A5/Z1/RUNG11/FP,XR
      COMMON/A9/EGO
      FP(1)=-2.0*Z1(3,J)*(XR(1)*XR(1)-((2.0*XR(5))/0.5)*(1.0-XR(1)-XR(
     12)-XR(4)))-Z1(4,J)*XR(1)-2.0*Z1(5,J)*XR(1)*XR(5)-Z1(6,J)*XR(1)*
     1XR(1)*2.0+Z1(7,J)*(2.0*XR(2)*XR(5)-(XR(1)*XR(6)/2.5))
     1-Z1(8,J)*(XR(2)*XR(1)-(XP(6)*(EGO-XR(1)-XR(2)-XR(4))/1.25))
```

```
1-Z1(3,J)*XR(4)*XR(1)
FP(2)=Z1(4,J)*XR(1)+Z1(5,J)*2.0*XR(1)*XR(5)+Z1(6,J)*XR(1)*XR(1)
1-Z1(7,J)*((2.0*XR(2)*XR(5))-(XR(1)*XR(6)/2.5))
1-Z1(8,J)*(XR(2)*XR(1)-((XR(6)/1.25)*(EGO-XR(1)-XR(2)-XR(4))))
1+Z1(9,J)*(EGO-XR(1)-XR(2)-XR(4))/2.0
FP(3)=2.0*Z1(5,J)*XR(1)*XR(5)+Z1(6,J)*XR(1)*XR(1)
FP(4)=Z1(9,J)*(EGO-XR(1)-XR(2)-XR(4))/2.0-Z1(3,J)*XR(4)*XR(1)
FP(5)=Z1(3,J)*(XR(1)*XR(1)-4.0*XR(5)*(EGO-XR(1)-XR(2)-XR(4)))-Z1
1(7,J)*(2.0*XR(2)*XR(5))-(XR(1)*XR(6)/2.5))-2.0*Z1(5,J)*XR(1)*XR(
15)
FP(6)=Z1(7,J)*(2.0*XR(2)*XR(5)-(XR(1)*XR(6)/2.5))+Z1(8,J)*(XR(2)
1*XR(1)-(XR(6)/1.25)*(EGO-XR(1)-XR(2)-XR(1)))
FP(7)=Z1(4,J)*XR(1)+Z1(3,J)*XR(1)*XR(4)
RETURN
END
C
C *******************************************************************
C THIS SUBROUTINE CALCULATES THE AMOUNT OF VOLATILE COMP-
C ONENTS(ETHYLENE GLYCOL,WATER,ACETALDEHYDE)FLASHED FROM
C THE REACTION MASS
      SUBROUTINE FLASH(DIST,NOX,T,J)
      DIMENSION XORG(7),XI(7),F(7),AK(7),FNEW(7)
      COMMON/R1/XI,XORG
      COMMON/F1/AK,F
      COMMON/FF1/PT,H
      FTOT=(XORG(1)+XORG(2)+XORG(3)+XORG(4))/2.0
      DO 701 I=5,7
      FTOT=FTOT+XORG(I)
701   CONTINUE
      DO 702 I=1,7
      F(I)=XORG(I)/FTOT
      AK(I)=0.0
702   CONTINUE
      AK(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      AK(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      AK(7)=(55.0*EXP(41.215329-(4860.8854/T)-(5.04843*ALOG(T)+
     1(3.05143E-17)*(T**6.0)))
      PT1=((F(5)*AK(5)+F(6)*AK(6)+F(7)*AK(7)))
      IF(PT1.LT.PT) GOTO 703
707   CALL CONV(ROOT1,NOX,J)
      IF(NOX.GT.10) GOTO 710
      DO 704 I=1,7
      FNEW(I)=F(I)/(1.0+ROOT1*((AK(I)*EXP(1.0+H)/PT)-1.0))
704   CONTINUE

      FNEWT=(1.0-ROOT1)*FTOT
      DIST=FTOT*ROOT1
      DO 705 I=1,7
      XI(I)=(FNEW(I)*FNEWT)
      IF(XI(I).GE.0.0) GOTO 705
705   CONTINUE
119   FORMAT(//)
      NOX=1
      IF((NOX-NOX).EQ.0) GOTO 706
      H=0
      GOTO 706
703   NOX=0
      GOTO 706
708   FORMAT(1X,'** VAC. APPLIED AT J=',I5,'**')
710   TYPE 711,J
711   FORMAT(1X,'* SOMETHING HAS GONE WRONG',/,'CHECK1 J=',I4)
706   IF(NOX.EQ.0) GOTO 709
```

```
709     RETURN
        END
C       *****************************************************************
C       THIS SUBROUTINE FINDS THE ROOT USING NEWTONS METHOD
        SUBROUTINE CONV(ROOT1,NOX,J)
        DIMENSION AK(7),F(7)
        COMMON/F1/AK,F
        COMMON/FF1/PT,H
        ROOT=0.00001
        ITN=0
35      FUNR=0.0
        DFUNR=0.0
        DO 1100 I=5,7
        FUNR=FUNR+(AK(I)*F(I))/(PT*(1.0+ROOT*((AK(I)/PT)-1.0)))
        DFUNR=DFUNR+(-AK(I)*F(I)*((AK(I)/PT)-1.0))/(PT*((1.0+ROOT*1
       1(AK(I)/PT)-1.0))**2))
1100    CONTINUE
        FUNR=FUNR-1.0
        IF(ABS(FUNR).LE.1.0E-06) GO TO 80
        IF(ITN.GT.20) GO TO 60
        ROOT=ROOT-FUNR/DFUNR
        ITN=ITN+1
        GO TO 35
60      TYPE 40
40      FORMAT(15X,'ITERATION DOES NOT CONVERGE')
80      IF(ROOT.GT.0.0)ROOT1=ROOT
        CONTINUE
        NOX=1
        RETURN
        END
C       *****************************************************************
C       THIS SUBROUTINE IS USED FOR SOLVING THE ADJOINT VARIABLE
C       EQUATIONS USING FOURTH-ORDER RUNGE-KUTTA METHOD
        SUBROUTINE RUNG2(ALPHA2,ALPHA3,A)
        DIMENSION X(7,2001),V(4,2001),Z(9,2001),ZI(9)
        DIMENSION VORG(4),VORG1(4),FP(4),VR(4)
        DIMENSION CP(4),CP1(4),CP2(4),CP3(4),CP4(4)
        DIMENSION VAP(7),TN(2001),EO(9),ZO(9),XB(9)
C       -------------------------------------------------------------
        COMMON/A2/X/A3/Z
        COMMON/E1/EO,ZO,RG
        COMMON/A6/TN/A7/V
        COMMON/FF1/PT,H
        COMMON/A9/EGO
        COMMON/RUNG21/FP,VR
C
        REAL MP
        REAL MPP
        A=A/5.0
        DO 200 JI=1,2000
        J1=2002-JI
        J=J1
        DO 55 I=1,4
        VORG1(I)=V(I,J)
55      CONTINUE
        N=1
        D=0.0
        DO 12 I=3,9
        ZI(I)=Z(I,J)
12      CONTINUE
        DO 398 I=1,7
```

```fortran
      XB(I)=X(I,J)
398   CONTINUE
4444  DO 201 N=1,4
      VORG(N)=V(N,J)
      VR(N)=V(N,J)
201   CONTINUE
      T=TN(J)-D*(TN(J)-TN(J-1))
      DO 397 I=1,7
      X(I,J)=X(I,J)-D*(X(I,J)-X(I,J-1))
397   CONTINUE
      DO 2333 I=3,9
      Z(I,J)=60.0*EXP(-EU(I)/(RG*T)+ZO(I))
2333  CONTINUE
      X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
      X3=(19.422+0.025*(T-413.0))/1000.0
      MPP=(X(1,J)+X(2,J)+X(3,J)+X(4,J))/2.0
      VTU=X1*MPP+X2*X(5,J)+X3*X(6,J)
      DO 22222 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 22222
      Z(I,J)=Z(I,J)/VTU
22222 CONTINUE
      D=D+0.2
      CALL EQTN2(ALPHA2,ALPHA3,J,T)
      DO 202 N=1,4
      CP1(N)=A*FP(N)
      VR(N)=VORG(N)-0.5*CP1(N)
202   CONTINUE
      CALL EQTN2(ALPHA2,ALPHA3,J,T)
      DO 203 N=1,4
      CP2(N)=A*FP(N)
      VR(N)=VORG(N)-0.5*CP2(N)
203   CONTINUE
      CALL EQTN2(ALPHA2,ALPHA3,J,T)
      DO 206 N=1,4
      CP3(N)=A*FP(N)
      VR(N)=VORG(N)-CP3(N)
206   CONTINUE
      CALL EQTN2(ALPHA2,ALPHA3,J,T)
      DO 205 N=1,4
      CP4(N)=A*FP(N)
      CP(N)=(CP1(N)+2.0*(CP2(N)+CP3(N))+CP4(N))/6.0
      V(N,J)=VORG(N)-CP(N)
205   CONTINUE
      N=N+1
      DO 7979 I=1,7
      X(I,J)=XB(I)
7979  CONTINUE
      IF(N.LE.5)GOTO 4444
      DO 56 N=1,4
      V(N,J-1)=V(N,J)
      V(N,J)=VORG1(N)
56    CONTINUE
      DO 578 I=3,9
      Z(I,J)=ZI(I)
578   CONTINUE
      DO 399 I=1,7
      X(I,J)=XB(I)
399   CONTINUE
200   CONTINUE
      A=A*5.0
```

```
      RETURN
      END
C-------------------------------------------------------------------
C     SUBROUTINE TO SOLVE ADJOINT EQUATIONS
      SUBROUTINE FQTN2(ALPHA2,ALPHA3,J,T)
      DIMENSION X(7,2001),Z(9,2001),FP(4),VR(4)
      DIMENSION VAP(7)
      COMMON/A2/X/A3/Z
      COMMON/A9/EGO
      COMMON/RUNG21/FP,VR
      COMMON/FF1/PT,H
C     CALCULATION OF THE FIRST DERIVATIVES OF "F" W.R.T."X"
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      P=PT/(2.0*(VAP(6)-PT))
      WX1=P;WX2=P;WX3=P;WX4=P
      Q=PT/(2.0*(VAP(5)-PT))
      GX1=Q;GX2=Q;GX3=Q;GX4=Q
      DF1X1=-4.0*Z(3,J)*X(1,J)+((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGO-X(1,J)
     1-X(2,J)-X(4,J))*GX1))-Z(4,J)-((2.0*Z(5,J))*(X(5,J)+X(1,J))+GX1
     1))-4.0*Z(6,J)*X(1,J)+2.0*Z(7,J)*X(2,J)*GX1-((Z(7,J)/2.5)*(X(1.
     1J)*WX1+X(6,J))-Z(8,J)*X(2,J)+((Z(8,J)/1.25)*(-X(6,J)+(EGO-X(
     11,J)-X(2,J)-X(4,J))*WX1))-Z(3,J)*X(4,J)
      DF1X2=((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGO-X(1,J)-X(2,J)-X(4,J))*GX2
     1))-2.0*Z(5,J)*X(1,J)*GX2+((2.0*Z(7,J))*(X(5,J)+X(2,J)*GX2))-
     1(Z(7,J)*X(1,J)/2.5)*WX2-Z(8,J)*X(1,J)+((Z(8,J)/1.25)*(-X(6,J)
     1+(EGO-X(1,J)-X(2,J)-X(4,J))*WX2))
      DF1X3=0.0
      DF1X4=((4.0*Z(3,J)/0.5)*(-X(5,J)+(EGO-X(1,J)-X(2,J)-X(4,J))*
     1GX4))-2.0*Z(5,J)*X(1,J)*GX4+2.0*Z(7,J)*X(2,J)*GX4-(Z(7,J)*X(1.
     1J)/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,J)+(EGO-X(1,J)-X(2,J)-X(4,J))
     1*WX4))-Z(3,J)*X(1,J)
      DF2X1=Z(4,J)+((2.0*Z(5,J))*(X(1,J)*GX1+X(5,J)))+2.0*Z(6,J)*X(1.J
     1)-2.0*Z(7,J)*X(2,J)*GX1+((Z(7,J)/2.5)*(X(1,J)*WX1+X(6,J))-Z(8,
     1J)*X(2,J)+((Z(8,J)/1.25)*(-X(6,J)+(EGO-X(1,J)-X(2,J)-X(4,J))*
     1WX1))-Z(9,J)/2.0
      DF2X2=2.0*Z(5,J)*X(1,J)*GX2-((2.0*Z(7,J))*(X(2,J)*GX2+X(5,J)))+
     1(Z(7,J)*X(1,J)/2.5)*WX2-Z(8,J)*X(1,J)+((Z(8,J)/1.25)*(-X(6,J)+
     1(EGO-X(1,J)-X(2,J)-X(4,J))*WX2))-Z(9,J)/2.0
      DF2X3=0.0
      DF2X4=2.0*Z(5,J)*X(1,J)*GX4-2.0*Z(7,J)*X(2,J)*GX4+(Z(7,J)*X(1,J)
     1/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,J)+(EGO-X(1,J)-X(2,J)-X(4,J))*
     1WX4))-Z(9,J)/2.0
      DF3X1=((2.0*Z(5,J))*(X(5,J)+X(1,J)*GX1))+2.0*Z(6,J)*X(1,J)
      DF3X2=2.0*Z(5,J)*X(1,J)*GX2
      DF3X3=0.0
      DF3X4=2.0*Z(5,J)*X(1,J)*GX4
      DF4X1=-Z(9,J)/2.0-Z(3,J)*X(4,J)
      DF4X2=-Z(9,J)/2.0
      DF4X3=0.0
      DF4X4=-Z(9,J)/2.0-Z(3,J)*X(1,J)
C-------------------------------------------------------------------
C     CALCULATION OF "FP"
      FP(1)=(2.0*ALPHA2*EGO*((EGO/X(1,J))-15.0))/(225.0*X(1,J)*X(1.J))
     1-(DF1X1*VR(1)+DF2X1*VR(2)+DF3X1*VR(3)+DF4X1*VR(4))
      FP(2)=-2.0*ALPHA3*X(2,J)-(DF1X2*VR(1)+DF2X2*VR(2)+DF3X2*VR(3)
     1+DF4X2*VR(4))
      FP(3)=-(DF1X3*VR(1)+DF2X3*VR(2)+DF3X3*VR(3)+DF4X3*VR(4))
      FP(4)=-2.0*ALPHA3*X(4,J)-(DF1X4*VR(1)+DF2X4*VR(2)+DF3X4*VR(3)
     1+DF4X4*VR(4))
      RETURN
```

```
      END
C----------------------------------------------------------------
C     SUBROUTINE FOR SOLVING THE "P" DIFFERENTIAL EQUAITONS USING
C     THE RUNGE-KUTTA METHOD
      SUBROUTINE RUNG3(ALPHA2,ALPHA3,A)
      DIMENSION X(7,2001),V(4,2001),VB(9),XB(9)
      DIMENSION Z(9,2001),HT2(101),HT1(101)
      DIMENSION P(4,4,101),PORG1(4,4),PR(4,4),PORG(4,4),TN(2001)
      DIMENSION CP1(4,4),CP2(4,4),CP3(4,4),CP4(4,4),CP(4,4)
      DIMENSION FP(4,4),R(1,4,101),DFTP(1,4,101)
      DIMENSION EO(9),ZO(9)
C----------------------------------------------------------------
      COMMON/A2/X/A3/Z/E1/EO,ZO,RG
      COMMON/A6/TN/A7/V/A18/HT2/A26/HT1
      COMMON/A9/EGO/FF1/PT,H/R1S/DFTP
      COMMON/A13/P/A14/PR/A21/FP/A16/R
C----------------------------------------------------------------
      REAL MP
      NNN=5
      A=A/FLOAT(NNN)
      DO 200 JI=1,100
      J1=102-JI
      J=J1
      DO 53 I=1,4
      DO 53 K=1,4
      PORG1(I,K)=P(I,K,J)
53    CONTINUE
      B=0.0
      M=1
      N=0
5555  CONTINUE
      DO 55 I=1,4
      DO 55 K=1,4
      PR(I,K)=P(I,K,J)
      PORG(I,K)=P(I,K,J)
55    CONTINUE
      L=((J-1)*20)-N+1
      DO 5959 I=1,7
      XB(I)=X(I,L)
      IF(I.GT.4)GOTO 5959
      VB(I)=V(I,L)
5959  CONTINUE
      T=TN(L)-B*(TN(L)-TN(L-1))
      DO 1 I=3,9
      Z(I,J)=60.0*EXP(-EO(I)/(RG*T)+ZO(I))
1     CONTINUE
      DO 4446 I=1,4
      X(I,L)=X(I,L)-B*(X(I,L)-X(I,L-1))
      IF(I.GT.4)GOTO 4446
      V(I,L)=V(I,L)-B*(V(I,L)-V(I,L-1))
4446  CONTINUE
      X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
      X3=(19.422+0.025*(T-413.0))/1000.0
      MP=(X(1,L)+X(2,L)+X(3,L)+X(4,L))/2.0
      VTO=X1*MP+X2*X(5,L)+X3*X(6,L)
      DO 2 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 2
      Z(I,J)=Z(I,J)/VTO
2     CONTINUE
      B=B+1.0/FLOAT(NNN)
```

```
      CALL EQTN3(ALPHA2,ALPHA3,J,L,T)
      DO 202 I=1,4
      DO 202 K=1,4
      CP1(I,K)=A*FP(I,K)
      PR(I,K)=PORG(I,K)-0.5*CP1(I,K)
202   CONTINUE
1111  CALL EQTN3(ALPHA2,ALPHA3,J,L,T)
      DO 203 I=1,4
      DO 203 K=1,4
      CP2(I,K)=A*FP(I,K)
      PR(I,K)=PORG(I,K)-0.5*CP2(I,K)
203   CONTINUE
1112  CALL EQTN3(ALPHA2,ALPHA3,J,L,T)
      DO 206 I=1,4
      DO 206 K=1,4
      CP3(I,K)=A*FP(I,K)
      PR(I,K)=PORG(I,K)-CP3(I,K)
206   CONTINUE
1113  CALL EQTN3(ALPHA2,ALPHA3,J,L,T)
      DO 205 I=1,4
      DO 205 K=1,4
      CP4(I,K)=A*FP(I,K)
      CP(I,K)=(CP1(I,K)+2.0*(CP2(I,K)+CP3(I,K))+CP4(I,K))/6.0
      P(I,K,J)=PORG(I,K)-CP(I,K)
205   CONTINUE
      DO 777 I=1,4
      DO 777 K=1,4
      PPP=P(I,K,J)
      IF(PPP.GT.1.0E+08)GOTO 779
777   CONTINUE
1114  M=M+1
      DO 9995 I=1,7
      X(I,L)=XB(I)
      IF(I.GT.4)GOTO 9995
      V(I,L)=VB(I)
9995  CONTINUE
      IF(M.LE.NNN)GOTO 5555
      N=N+1
      L=0.0
      M=1
      IF(N.LE.19)GOTO 5555
311   FORMAT(1X,'J=',I3/1X,'P(I,K,J)=',16F15.8)
      DO 56 I=1,4
      DO 56 K=1,4
      P(I,K,J-1)=P(I,K,J)
      P(I,K,J)=PORG1(I,K)
56    CONTINUE
200   CONTINUE
779   A=A*FLOAT(NNN)
      RETURN
      END
C-----------------------------------------------------------------
C        SUBROUTINE TO SOLVE "P" DIFFERENTIAL EQUATIONS
      SUBROUTINE EQTN3(ALPHA2,ALPHA3,J,L,T)
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001),EQ(9),ZO(9)
      DIMENSION PR(4,4),FP(4,4)
      DIMENSION R(1,4,101),RR(1,4),TRR(4,1)
      DIMENSION HT2(101),RMR(4,4),HT1(101),DFTP(1,4,101)
      DIMENSION FX(4,4),XF(4,4),PFX(4,4),PXF(4,4)
      DIMENSION SDHX(4,4)
```

```
C
      COMMON/A2/X/A3/Z/E1/E0,Z0,RG
      COMMON/A7/V
      COMMON/A9/EGO
      COMMON/A14/PR
      COMMON/A16/R/A17/RR,TRR
      COMMON/A18/HT2/A19/FX,XF
      COMMON/A20/SDHX/A21/FP/FF1/PT,H
      COMMON/A26/HT1/B15/DFTP
      CALL RICCAT(J,L,T)
      CALL HAMLT2(J,L,T)
      DO 324 I1=1,4
      TRR(I1,1)=TRR(I1,1)*HT2(J)
324   CONTINUE
C     MULTIPLICATION OF THE MATRICES "RR" AND "TRR"
      J2=1
      DO 325 K2=1,4
      DO 325 L2=1,4
      RMR(K2,L2)=TRR(K2,J2)*RR(J2,L2)
325   CONTINUE
C
      CALL FXF(J,L,T)
C     MULTIPLICATION OF THE MATRICES "FX" AND "P"
1188  DO 318 J2=1,4
      DO 318 K2=1,4
      PFX(J2,K2)=0.0
      DO 316 L2=1,4
      PFX(J2,K2)=PFX(J2,K2)+FX(J2,L2)*PR(L2,K2)
316   CONTINUE
318   CONTINUE
C
C     MULTIPLICATION OF THE MATRICES "XF" AND "P"
1189  DO 418 J2=1,4
      DO 418 K2=1,4
      PXF(J2,K2)=0.0
      DO 416 L2=1,4
      PXF(J2,K2)=PXF(J2,K2)+PR(J2,L2)*XF(L2,K2)
416   CONTINUE
418   CONTINUE
C
888   CALL HAMLTX(ALPHA2,ALPHA3,J,L,T)
      DO 496 I=1,4
      DO 496 K=1,4
      FP(I,K)=-SDHX(I,K)-PXF(I,K)-PFX(I,K)+RMR(T,K)
496   CONTINUE
      RETURN
      END
C
C     THIS SUBROUTINE CALCULATES THE ELEMENTS IN THE RICCATI MATRIX
C     "R" FOR USE IN THE SOLUTION OF THE "P" MATRIX
      SUBROUTINE RICCAT(J,L,T)
      DIMENSION X(7,2001),V(4,2001),ZP(9,101),ZN(9)
      DIMENSION Z(9,2001),E0(9),Z0(9)
      DIMENSION PR(4,4)
      DIMENSION DF1TX(4),DF2TX(4),DF3TX(4),DF4TX(4)
      DIMENSION FTX(1,4),DFT(1,4),DFTP(1,4,101),HT1(101)
      DIMENSION FTP(1,4),RR(1,4),R(1,4,101),TRR(4,1)
      DIMENSION VAP(7)
C
      COMMON/A2/X/A3/Z
```

```
      COMMON/A7/V/E1/EO,ZO,RG
      COMMON/A9/EGO
      COMMON/A14/PR
      COMMON/A15/DFT/A16/R/A17/RR,TPR
      COMMON/FE1/PT,H/B15/DFTP/A26/HT1
C     CALCULATION OF THE ELEMENTS IN THE MATRIX OBTAINED BY DIFFEREN-
C     TIATING THE HAMILTONIAN W.R.T. "T" AND "X"
      REAL MP
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      P=PT/(2.0*(VAP(6)-PT))
      WX1=P;WX2=P;WX3=P;WX4=P
      Q=PT/(2.0*(VAP(5)-PT))
      GX1=0;GX2=0;GX3=0;GX4=0
      A=VAP(5)*((3729.0/(T*T))-4.042/T)
      B=VAP(6)*(1757.853/(T-33.274)**2)
      C=-(X(5,L)*A+X(6,L)*B)
      GT=C/(VAP(5)-PT)
      WT=C/(VAP(6)-PT)
      D=-(PT*A)/(2.0*((VAP(5)-PT)**2))
      E=-(PT*B)/(2.0*((VAP(6)-PT)**2))
      GX1T=D;GX2T=D;GX3T=D;GX4T=D
      WX1T=E;WX2T=E;WX3T=E;WX4T=E
C------------------------------------------------------------
      X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
      X3=(19.422+0.025*(T-413.0))/1000.0
      MP=(X(1,L)+X(2,L)+X(3,L)+X(4,L))/2.0
      VP=(((MP)*191.5*0.0014+X(5,L)*60.6*0.0014+X(6,L)*0.025)/1000.0)
     1+X2*GT+X3*WT
      VTO=X1*MP+X2*X(5,L)+X3*X(6,L)
      DO 1 I=3,9
      ZN(I)=60.0*EXP(-EO(I)/(RG*T))+ZO(I)
      ZP(I,J)=EO(I)*ZN(I)/(RG*T*T)
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 1
      ZP(I,J)=(ZP(I,J)/VTO)-((ZN(I)/(VTO*VTO))*VP)
    1 CONTINUE
      DF1TX(1)=(-4.0*X(1,L)-X(4,L))*ZP(3,J)-(4.0/0.5)*(Z(3,J)*GT+X(5.
     1L)*ZP(3,J))+(4.0/0.5)*(EGO-X(1,L)-X(2,L)-X(4,L))*(Z(3,J)*GX1T+
     1GX1*ZP(3,J))-ZP(4,J)-2.0*X(1,L)*(Z(5,J)*GX1T+GX1*ZP(5,J))-2.0*
     1(Z(5,J)*GT+X(5,L)*ZP(5,J))-4.0*X(1,L)*ZP(6,J)+2.0*X(2,L)*(Z(7,J
     1)*GX1T+GX1*ZP(7,J))-(X(1,L)/2.5)*(Z(7,J)*WX1T+WX1*ZP(7,J))-(Z(7
     1,J)*WT+X(6,L)*ZP(7,J))/2.5-X(2,L)*ZP(8,J)-(Z(8,J)*WT+X(6,L)*ZP
     1(8,J))/1.25+((EGO-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*WX1T+
     1WX1*ZP(8,J))
      DF1TX(2)=(-4.0/0.5)*(Z(3,J)*GT+X(5,L)*ZP(3,J))+(4.0*(EGO-X(1,L)-
     1X(2,L)-X(4,L))/0.5)*(Z(3,J)*GX2T+GX2*ZP(3,J))-2.0*X(1,L)*(Z(5,J
     1)*GX2T+GX2*ZP(5,J))+2.0*X(2,L)*(Z(7,J)*GX2T+GX2*ZP(7,J))+2.0*(Z
     1(7,J)*GT+X(5,L)*ZP(7,J))-(X(1,L)/2.5)*(Z(7,J)*WX2T+WX2*ZP(7,J))
     1-X(1,L)*ZP(8,J)-(Z(8,J)*WT+X(6,L)*ZP(8,J))/1.25+((EGO-X(1,L)-X(
     12,L)-X(4,L))/1.25)*(Z(8,J)*WX2T+WX2*ZP(8,J))
      DF1TX(3)=0.0
      DF1TX(4)=(-4.0/0.5)*(Z(3,J)*GT+X(5,L)*ZP(3,J))+(4.0*(EGO-X(1,L)-
     1X(2,L)-X(4,L))/0.5)*(Z(3,J)*GX4T+GX4*ZP(3,J))-2.0*X(1,L)*(Z(5,J
     1)*GX4T+GX4*ZP(5,J))+2.0*X(2,L)*(Z(7,J)*GX4T+GX4*ZP(7,J))-(X(1,L
     1)/2.5)*(Z(7,J)*WX4T+WX4*ZP(7,J))-(Z(8,J)*WT+X(6,L)*ZP(8,J))/
     11.25+((EGO-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*WX4T+WX4*ZP(8,
     1J))-X(1,L)*ZP(3,J)
C------------------------------------------------------------
      DF2TX(1)=ZP(4,J)+2.0*X(1,L)*(Z(5,J)*GX1T+GX1*ZP(5,J))+2.0*(Z(5.
     1J)*GT+X(5,L)*ZP(5,J))+4.0*X(1,L)*ZP(6,J)-2.0*X(2,L)*(Z(7,J)*
```

```
      1GX1T+GX1*ZP(7,J))+(X(1,L)/2.5)*(Z(7,J)*WX1T+WX1*ZP(7,J))+(Z(7
      1,J)*WT+X(6,L)*ZP(7,J))/2.5-X(2,L)*ZP(8,J)-(Z(8,J)*WT+X(6,L)*
      1ZP(8,J))/1.25+((EGO-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*WX1T
      1+WX1*ZP(8,J))-ZP(9,J)/2.0
C
      DF2TX(2)=2.0*X(1,L)*(Z(5,J)*GX2T+GX2*ZP(5,J))-2.0*X(2,L)*(Z(7,
      1J)*GX2T+GX2*ZP(7,J))-2.0*(Z(7,J)*GT+X(5,L)*ZP(7,J))+(X(1,L)/
      12.5)*(Z(7,J)*WX2T+WX2*ZP(7,J))-X(1,L)*ZP(8,J)-(Z(8,J)*WT+X(6,
      1L)*ZP(8,J))/1.25+((EGO-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*
      1WX2T+WX2*ZP(8,J))-ZP(9,J)/2.0
      DF2TX(3)=0.0
      DF2TX(4)=2.0*X(1,L)*(Z(5,J)*GX4T+GX4*ZP(5,J))-2.0*X(2,L)*(Z(7,
      1J)*GX4T+GX4*ZP(7,J))+(X(1,L)/2.5)*(Z(7,J)*WX4T+WX4*ZP(7,J))-
      1(Z(8,J)*WT+X(6,L)*ZP(8,J))/1.25+((EGO-X(1,L)-X(2,L)-X(4,L))/
      11.25)*(Z(8,J)*WX4T+WX4*ZP(8,J))-ZP(9,J)/2.0
C
      DF3TX(I)=2.0*X(1,L)*(Z(5,J)*GX1T+GX1*ZP(5,J))+2.0*(Z(5,J)*GT+
      1X(5,L)*ZP(5,J))+4.0*X(1,L)*ZP(6,J)
      DF3TX(2)=2.0*X(1,L)*(Z(5,J)*GX2T+GX2*ZP(5,J))
      DF3TX(3)=0.0
      DF3TX(4)=2.0*X(1,L)*(Z(5,J)*GX4T+GX4*ZP(5,J))
C
      DF4TX(1)=-ZP(9,J)/2.0-X(4,L)*ZP(3,J)
      DF4TX(2)=-ZP(9,J)/2.0
      DF4TX(3)=0.0
      DF4TX(4)=-ZP(9,J)/2.0-X(1,L)*ZP(3,J)
C
C
      DO 70 M=1,4
      DO 70 I=1,4
      GOTO (71,72,73,74),M
71    DF1TX(I)=V(M,L)*DF1TX(I)
      GOTO 70
72    DF2TX(I)=V(M,L)*DF2TX(I)
      GOTO 70
73    DF3TX(I)=V(M,L)*DF3TX(I)
      GOTO 70
74    DF4TX(I)=V(M,L)*DF4TX(I)
      GOTO 70
70    CONTINUE
      DO 77 I=1,4
      FTX(1,I)=DF1TX(I)+DF2TX(I)+DF3TX(I)+DF4TX(I)
77    CONTINUE
C
1111  CALL FTEMP(J,L,T)
C     MULTIPLICATION OF THE MATRICES "DFT" AND "P"
      J2=1
      DO 319 K2=1,4
      FTP(J2,K2)=0.0
      DO 320 L2=1,4
      FTP(J2,K2)=FTP(J2,K2)+DFT(J2,L2)*PR(L2,K2)
320   CONTINUE
319   CONTINUE
C
1112  DO 321 I=1,4
      RR(1,I)=FTX(1,I)+FTP(1,I)
321   CONTINUE
      DO 325 I=1,4
      R(1,I,J)=RR(1,I)
325   CONTINUE
C
```

```fortran
C     TRANSPOSE OF THE "RR" MATRIX INTO "TRR" MATRIX
1113  K1=1
      DO 322 I1=1,4
      TRR(I1,K1)=RR(K1,I1)
322   CONTINUE
1114  RETURN
      END
C------------------------------------------------------------------------
C     THIS SUBROUTINE CALCULATES THE DERIVATIVES OF "F" W.R.T."T"
C     AND STORES THE DERIVATIVE VALUES OF THE HAMILTONIAN W.R.T."T"
      SUBROUTINE FTEMP(J,L,T)
      DIMENSION X(7,2001),V(4,2001),ZP(9,101),ZN(9)
      DIMENSION Z(9,2001)
      DIMENSION DFT(1,4),VAP(7)
      DIMENSION HT1(101),DFTP(1,4,101),E0(9),Z0(9)
C------------------------------------------------------------------------
      COMMON/A2/X/A3/Z
      COMMON/A7/V/E1/F0,Z0,RG
      COMMON/A9/EG0
      COMMON/A15/DFT/A26/HT1/FF1/PT,H/B15/DFTP
C     CALCULATION OF THE DERIVATIVES OF "F" W.R.T."T"
      REAL MP
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      A=VAP(5)*((3729.0/(T*T))-4.042/T)
      B=VAP(6)*(1757.853/(T-33.274)**2)
      C=-(X(5,L)*A+X(6,L)*B)
      GT=C/(VAP(5)-PT)
      WT=C/(VAP(6)-PT)
C------------------------------------------------------------------------
      X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
      X3=(19.422+0.025*(T-413.0))/1000.0
      MP=(X(1,L)+X(2,L)+X(3,L)+X(4,L))/2.0
      VP=(((MP)*191.5*0.0014+X(5,L)*60.6*0.0014+X(6,L)*0.025)/1000.0)
     1+X2*GT+X3*WT
      VTO=X1*MP+X2*X(5,L)+X3*X(6,L)
      DO 1 I=3,9
      ZN(I)=60.0*EXP(-E0(I)/(RG*T)+Z0(I))
      ZP(I,J)=E0(I)*ZN(I)/(RG*T*T)
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 1
      ZP(I,J)=(ZP(I,J)/VTO)-((ZN(I)/(VTO*VTO))*VP)
1     CONTINUE
      DFT(1,1)=-2.0*X(1,L)*X(1,L)*ZP(3,J)+(4.0*(EG0-X(1,L)-X(2,L)-X(4,
     1L))/0.5)*(Z(3,J)*GT+X(5,L)*ZP(3,J))-ZP(4,J)*X(1,L)*ZP(6,J)+2.0*X(2,L)*
     1(Z(5,J)*GT+X(5,L)*ZP(5,J))-2.0*X(1,L)*X(1,L)*ZP(6,J)+2.0*X(2,L)
     1*(Z(7,J)*GT+X(5,L)*ZP(7,J))-(X(1,L)/2.5)*(Z(7,J)*WT+X(6,L)*ZP(7
     1,J))-X(2,L)*X(1,L)*ZP(8,J)+((EG0-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z
     1(8,J)*WT+X(6,L)*ZP(8,J))-X(4,L)*X(1,L)*ZP(3,J)
      DFT(1,2)=X(1,L)*ZP(4,J)+2.0*X(1,L)*(Z(5,J)*GT+X(5,L)*ZP(5,J))
     1+2.0*X(1,L)*X(1,L)*ZP(6,J)-2.0*X(2,L)*(Z(7,J)*GT+X(5,L)*ZP(7,
     1J))+(X(1,L)/2.5)*(Z(7,J)*WT+X(6,L)*ZP(7,J))-X(2,L)*X(1,L)*ZP(
     1(8,J)+((EG0-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*WT+X(6,L)*
     1ZP(8,J))+((EG0-X(1,L)-X(2,L)-X(4,L))/2.0)*ZP(9,J)
      DFT(1,3)=2.0*X(1,L)*(Z(5,J)*GT+X(5,L)*ZP(5,J))+2.0*X(1,L)*X(1,L)
     1*ZP(6,J)
      DFT(1,4)=((EG0-X(1,L)-X(2,L)-X(4,L))/2.0)*ZP(9,J)-X(4,L)*X(1,L)
     1*ZP(3,J)
C------------------------------------------------------------------------
      DO 3333 I=1,4
      DFTP(1,I,J)=DFT(1,I)
```

```fortran
3333    CONTINUE
        SUM=0.0
        DO 19 I=1,4
        SUM=SUM+V(I,L)*DFT(1,I)
19      CONTINUE
        HT1(J)=SUM
1111    RETURN
        END
C
C       THIS SUBROUTINE CALCULATES THE SECOND DERIVATIVES OF THE HAMILTO
C       -NIAN W.R.T."T"
        SUBROUTINE HAMLT2(J,L,T)
        DIMENSION X(7,2001),V(4,2001),ZP(9,101),HP(9,101),ZN(9)
        DIMENSION Z(9,2001),EO(9),ZO(9)
        DIMENSION HT2(101),FT2(4),ZP1(9)
        DIMENSION VAP(7)
C
        COMMON/A2/X/A3/Z
        COMMON/A7/V/E1/EO,ZO,RG
        COMMON/A9/EGO
        COMMON/A18/HT2/FF1/PT,H
C       CALCULATION OF THE SECOND DERIVATIVES OF "F" W.R.T."T"
        REAL MP
        VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
        VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
        A=VAP(5)*((3729.0/(T*T))-4.042/T)
        B=VAP(6)*(1757.853/(T-33.274)**2)
        C=-(X(5,L)*A+X(6,L)*B)
        GT=C/(VAP(5)-PT)
        WT=C/(VAP(6)-PT)
        G=VAP(5)*((-2.0*3729.0/(T**3))+(4.042/(T**2)))+((3729.0/(T**2))
       1-(4.042/T))*A
        W=VAP(6)*(-2.0*1757.853/((T-33.274)**3))+(1757.853/((T-33.274)
       1**2))*B
        GT2=(-X(5,L)*G-X(6,L)*W-WT*B-2.0*GT*A)/(VAP(5)-PT)
        WT2=(-X(5,L)*G-X(6,L)*W-GT*A-2.0*WT*B)/(VAP(6)-PT)
C
5555    X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
        X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
        X3=(19.422+0.025*(T-413.0))/1000.0
        MP=(X(1,L)+X(2,L)+X(3,L)+X(4,L))/2.0
        VP=(((MP)*191.5*0.0014+X(5,L)*60.6*0.0014+X(6,L)*0.025)/1000.0)
       1+X2*GT+X3*WT
        VTO=X1*MP+X2*X(5,L)+X3*X(6,L)
        DO 1 I=3,9
        ZN(I)=60.0*EXP(-EO(I)/(RG*T)+ZO(I))
        ZP(I,J)=EO(I)*ZN(I)/(RG*T*T)
        IF((I.EQ.4).OR.(I.EQ.9))GOTO 1
        ZP(I,J)=(ZP(I,J)/VTO)-((ZN(I)/(VTO*VTO))*VP)
1       CONTINUE
        DO 2 I=3,9
        ZP1(I)=EO(I)*ZN(I)/(RG*T*T)
        IF((I.EQ.4).OR.(I.EQ.9))GOTO 3
        HP(I,J)=(ZP1(I)/(VTO*T*T))*((EO(I)/RG)-(2.0*T+(T*T*VP/VTO)))
       1-(VP/(VTO*VTO))*(ZP1(I)-((2.0*ZN(I)*VP)/VTO))-(ZN(I)/(VTO
       1*VTO))*(X2*GT2+X3*WT2+2.0*GT*((60.6*0.0014)/1000.0)+2.0*
       1WT*(0.025/1000.0))
       11000.0))
        GOTO 2
3       HP(I,J)=(ZP1(I)/T)*((EO(I)/(RG*T))-2.0)
        GOTO 2
```

```
2     CONTINUE
4444  FT2(1)=-2.0*X(1,L)*X(1,L)*HP(3,J)+4.0*((EGO-X(1,L)-X(2,L)-X(4,
     1,L))/0.5)*(Z(3,J)*GT2+X(5,L)*HP(3,J)+2.0*GT*ZP(3,J))-X(1,L)*
     1HP(4,J)-2.0*X(1,L)*(Z(5,J)*GT2+X(5,L)*HP(5,J)+2.0*GT*ZP(5,J))
     1-2.0*X(1,L)*X(1,L)*HP(6,J)+2.0*X(2,L)*(Z(7,J)*GT2+X(5,L)*HP(
     17,J)+2.0*GT*ZP(7,J))-(X(1,L)/2.5)*(Z(7,J)*WT2+X(6,L)*HP(7,J)
     1+2.0*WT*ZP(7,J))-X(2,L)*X(1,L)*HP(8,J)+((EGO-X(1,L)-X(2,L)-X(4,
     1,L))/1.25)*(Z(8,J)*WT2+X(6,L)*HP(8,J)+2.0*WT*ZP(8,J))-X(4,L)*
     1X(1,L)*HP(3,J)
      FT2(2)=X(1,L)*HP(4,J)+2.0*X(1,L)*(Z(5,J)*GT+ZP(5,J)*GT+X(5,L)*
     1ZP(5,J)+X(5,L)*HP(5,J))+2.0*X(1,L)*X(1,L)*HP(6,J)-2.0*X(2,L)*
     1(Z(7,J)*GT2+X(5,L)*HP(7,J)+2.0*GT*ZP(7,J))+(X(1,L)/2.5)*(Z(7,
     1J)*WT2+X(6,L)*HP(7,J)+2.0*WT*ZP(7,J))-X(2,L)*X(1,L)*HP(8,J)+
     1((EGO-X(1,L)-X(2,L)-X(4,L))/1.25)*(Z(8,J)*WT2+X(6,L)*HP(8,L)
     1+2.0*WT*ZP(8,J))+(EGO-X(1,L)-X(2,L)-X(4,L))/2.0)*HP(9,J)
      FT2(3)=2.0*X(1,L)*(Z(5,J)*GT2+X(5,L)*HP(5,J)+2.0*GT*ZP(5,J))
     1+2.0*X(1,L)*X(1,L)*HP(6,J)
      FT2(4)=((EGO-X(1,L)-X(2,L)-X(4,L))*HP(9,J))/2.0-X(4,L)*X(1,L)*HP
     1(3,J)
C
1119  SUM=0.0
      DO 323 I=1,4
      SUM=SUM+V(I,L)*FT2(I)
323   CONTINUE
      HT2(J)=SUM
      HT2(J)=1.0/HT2(J)
1111  RETURN
      END
C
C     THIS SUBROUTINE CALCULATES THE DERIVATIVES OF "F" W.R.T."X"
      SUBROUTINE FXF(J,L,T)
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001)
      DIMENSION FX(4,4),XF(4,4)
      DIMENSION VAP(7)
C
      COMMON/A2/X/A3/Z
      COMMON/A7/V
      COMMON/A9/EGO
      COMMON/A19/FX,XF
      COMMON/FF1/PT,H
C
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      P=PT/(2.0*(VAP(6)-PT))
      WX1=P;WX2=P;WX3=P;WX4=P
      Q=PT/(2.0*(VAP(5)-PT))

      GX1=Q;GX2=Q;GX3=Q;GX4=Q
C
      FX(1,1)=-4.0*Z(3,J)*X(1,L)+((4.0*Z(3,J)/0.5)*(-X(5,L)+(EGO-X(1,
     1L)-X(2,L)-X(4,L))*GX1))-Z(4,J)-((2.0*Z(5,J))*(X(5,L)+X(1,L)*
     1GX1))-4.0*Z(6,J)*X(1,L)+2.0*Z(7,J)*X(2,L)*GX1-((Z(7,J)/2.5)*
     1(X(1,L)*WX1+X(6,L)))-Z(8,J)*X(2,L)+((Z(8,J)/1.25)*(-X(6,L)+(
     1EGO-X(1,L)-X(2,L)-X(4,L))*WX1))-Z(3,J)*X(4,L)
      FX(2,1)=((4.0*Z(3,J)/0.5)*(-X(5,L)+(EGO-X(1,L)-X(2,L)-X(4,L))*
     1GX2))-2.0*Z(5,J)*X(1,L)*WX2+((2.0*Z(7,J)*(X(5,L)+X(2,L)*GX2))
     1-(Z(7,J)*X(1,L)/2.5)*WX2-Z(8,J)*X(1,L)+((Z(8,J)/1.25)*(-X(6,L)
     1+(EGO-X(1,L)-X(2,L)-X(4,L))*WX2))
      FX(3,1)=0.0
      FX(4,1)=((4.0*Z(3,J)/0.5)*(-X(5,L)+(EGO-X(1,L)-X(2,L)-X(4,L))*
```

```fortran
1GX4))-2.0*Z(5,J)*X(1,L)*GX4+2.0*Z(7,J)*X(2,L)*GX4-(Z(7,J)*X(1,
1L)/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,L)+(EGO-X(1,L)-X(2,L)-X(4,L))
1*WX4))-Z(3,J)*X(1,L)
FX(1,2)=Z(1,J)+((2.0*Z(5,J))*(X(1,L)*GX1+X(5,L)))+2.0*Z(6,J)*X(1
1,L)-2.0*Z(7,J)*X(2,L)*GX1+((Z(7,J)/2.5)*(X(1,L)*WX1+X(6,L)))-
1Z(8,J)*X(2,L)+((Z(8,J)/1.25)*(-X(6,L)+(EGO-X(1,L)-X(2,L)-X(4,L)
1)*WX1))-Z(9,J)/2.0
FX(2,2)=2.0*Z(5,J)*X(1,L)*GX2-((2.0*Z(7,J))*(X(2,L)*GX2+X(5,L)))
1+(Z(7,J)*X(1,L)/2.5)*WX2-Z(8,J)*X(1,L)+((Z(8,J)/1.25)*(-X(6,L)
1+(EGO-X(1,L)-X(2,L)-X(4,L))*WX2))-Z(9,J)/2.0
FX(1,3)=((2.0*Z(5,J))*(X(5,L)+X(1,L)*GX1))+4.0*Z(6,J)*X(1,L)
FX(2,3)=2.0*Z(5,J)*X(1,L)*GX2
FX(3,3)=0.0
FX(4,3)=2.0*Z(5,J)*X(1,L)*GX4
FX(1,4)=-Z(9,J)/2.0-Z(3,J)*X(4,L)
FX(2,4)=-Z(9,J)/2.0
FX(3,4)=0.0
FX(4,4)=-Z(9,J)/2.0-Z(3,J)*X(1,L)
FX(3,2)=0.0
FX(4,2)=2.0*Z(5,J)*X(1,L)*GX4-2.0*Z(7,J)*X(2,L)*GX4+(Z(7,J)*X(
11,L)/2.5)*WX4+((Z(8,J)/1.25)*(-X(6,L)+(EGO-X(1,L)-X(2,L)-X(4,L
1))*WX4))-Z(9,J)/2.0
C------------------------------------------------------------------
C     TRANSPOSE OF THE "FX" MATRIX INTO "XF" MATRIX
      DO 301 K1=1,4
      DO 301 I1=1,4
      XF(I1,K1)=FX(K1,I1)
301   CONTINUE
      RETURN
      END
C------------------------------------------------------------------
C     THIS SUBROUTINE CALCULATES THE SECOND DERIVATIVES OF THE HAMIL-
C     TONIAN W.R.T."X"
      SUBROUTINE HAMLTX(ALPHA2,ALPHA3,J,L,T)
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001),SDJX(4,4)
      DIMENSION F1X2(4,4),F2X2(4,4),F3X2(4,4),F4X2(4,4)
      DIMENSION FX2(4,4),SDHX(4,4)
      DIMENSION VAP(7)
C------------------------------------------------------------------
      COMMON/A2/X/A3/Z/A7/V/A9/EGO
      COMMON/A20/SDHX/FF1/PT,H
C------------------------------------------------------------------
C     CALCULATION OF THE SECOND DERIVATIVES OF "J" IN THE HAMILTONIAN
C     W.R.T."X"
      DO 10 I=1,4
      DO 10 K=1,4
      SDJX(I,K)=0.0
10    CONTINUE
      DO 11 I=1,4
      GOTO (21,22,23,24),I
21    SDJX(I,I)=(2.0*ALPHA2*EGO*(3.0*EGO/X(1,L)-2.0*15.0))/(225.0*X(1,
1L)*X(1,L)*X(1,L))+SDJX(I,I)
      GOTO 11
22    SDJX(I,I)=2.0*ALPHA3
      GOTO 11
23    SDJX(I,I)=0.0
      GOTO 11
24    SDJX(I,I)=2.0*ALPHA3
11    CONTINUE
```

```
C-----------------------------------------------------------------
C     CALCULATION OF THE SECOND DERIVATIVES OF "F" IN THE HAMILTONIAN
C     W.R.T. "X"
      VAP(5)=10.0**(21.61-3729.0/T-4.042*ALOG10(T))/760.0
      VAP(6)=10.0**(8.064103-1757.853/(T-33.274))/760.0
      P=PT/(2.0*(VAP(6)-PT))
      WX1=P;WX2=P;WX3=P;WX4=P
      Q=PT/(2.0*(VAP(5)-PT))
      GX1=Q;GX2=Q;GX3=Q;GX4=Q
C-----------------------------------------------------------------
      F1X2(1,1)=-4.0*Z(3,J)-4.0*Z(6,J)-(4.0*Z(3,J)/0.5+4.0*Z(5,J))+
     1GX1-((2.0*Z(7,J)/2.5)+(2.0*Z(8,J)/1.25))*WX1
      F1X2(1,2)=((-2.0*Z(3,J)/0.5)+(2.0*Z(7,J)))*GX1-((2.0*Z(3,J)/0.5
     1)+(2.0*Z(5,J)))*GX2-((Z(7,J)/2.5)+(Z(8,J)/1.25))*WX2-Z(6,J)-(Z
     1(8,J)/1.25)*WX1
      F1X2(1,3)=0.0
      F1X2(1,4)=(-2.0*Z(3,J)/0.5)*GX1-((2.0*Z(3,J)/0.5)+2.0*Z(5,J))*
     1GX4-(Z(8,J)/1.25)*WX1-((Z(7,J)/2.5)+(Z(8,J)/1.25))*WX4-Z(3,J)
      F1X2(2,1)=F1X2(1,2)
      F1X2(2,2)=((-4.0*Z(3,J)/0.5)+4.0*Z(7,J))*GX2-(2.0*Z(8,J)/1.25)*
     1WX2
      F1X2(2,3)=0.0
      F1X2(2,4)=(-2.0*Z(3,J)/0.5)*GX2-((2.0*Z(3,J)/0.5)-2.0*Z(7,J))*
     1GX4-(Z(8,J)/1.25)*WX2-(Z(8,J)/1.25)*WX4
      DO 29 I=1,4
      F1X2(3,I)=0.0
29    CONTINUE
      F1X2(4,1)=F1X2(1,4)
      F1X2(4,2)=F1X2(2,4)
      F1X2(4,3)=0.0
      F1X2(4,4)=(-4.0*Z(3,J)/0.5)*GX4-(2.0*Z(8,J)/1.25)*WX4
C-----------------------------------------------------------------
      F2X2(1,1)=4.0*Z(5,J)*GX1+4.0*Z(6,J)+(2.0*Z(7,J)/2.5)*WX1-(2.0*
     1Z(8,J)/1.25)*WX1
      F2X2(1,2)=2.0*Z(5,J)*GX2-2.0*Z(7,J)*GX1+((Z(7,J)/2.5)-(Z(8,J)/
     11.25))*WX2-Z(8,J)-(Z(8,J)/1.25)*WX1
      F2X2(1,3)=0.0
      F2X2(1,4)=2.0*Z(5,J)*GX4+((Z(7,J)/2.5)-(Z(8,J)/1.25))*WX4-(Z(8,J
     1)/1.25)*WX1
      F2X2(2,1)=F2X2(1,2)
      F2X2(2,2)=-4.0*Z(7,J)*GX2-(2.0*Z(8,J)/1.25)*WX2
      F2X2(2,3)=0.0
      F2X2(2,4)=-2.0*Z(7,J)*GX4-(Z(8,J)/1.25)*WX2-(Z(8,J)/1.25)*WX4
      DO 30 I=1,4
      F2X2(3,I)=0.0
30    CONTINUE
      F2X2(4,1)=F2X2(1,4)
      F2X2(4,2)=F2X2(2,4)
      F2X2(4,3)=0.0
      F2X2(4,4)=(2.0*Z(8,J)/1.25)*WX4
C-----------------------------------------------------------------
      F3X2(1,1)=4.0*Z(5,J)*GX1+4.0*Z(6,J)
      F3X2(1,2)=2.0*Z(5,J)*GX2
      F3X2(1,3)=0.0
      F3X2(1,4)=2.0*Z(5,J)*GX4
      F3X2(2,1)=F3X2(1,2)
      F3X2(2,2)=0.0;F3X2(2,3)=0.0;F3X2(2,4)=0.0
      DO 32 I=1,4
      F3X2(3,I)=0.0
32    CONTINUE
      F3X2(4,1)=F3X2(1,4)
```

```
      F3X2(4,2)=F3X2(2,4)
      F3X2(4,3)=0.0
      F3X2(4,4)=0.0
C-----------------------------------------------------------
      DO 34 I=1,4
      DO 34 K=1,4
      IF((I.EQ.1).AND.(K.EQ.4))GOTO 34
      IF((I.EQ.4).AND.(K.EQ.1))GOTO 34
      F4X2(I,K)=0.0
 34   CONTINUE
      F4X2(1,4)=-Z(3,J)
      F4X2(4,1)=F4X2(1,4)
C-----------------------------------------------------------
C
C     MULTIPLICATION OF THE ADJOINT VARIABLES AND THE SECOND DERIVATI-
C     VES OF "F" W.P.T. "X"
      DO 42 M=1,4
      DO 42 I=1,4
      DO 42 K=1,4
      GOTO(43,44,45,46),M
 43   F1X2(I,K)=V(M,L)*F1X2(I,K)
      GOTO 42
 44   F2X2(I,K)=V(M,L)*F2X2(I,K)
      GOTO 42
 45   F3X2(I,K)=V(M,L)*F3X2(I,K)
      GOTO 42
 46   F4X2(I,K)=V(M,L)*F4X2(I,K)
      GOTO 42
 42   CONTINUE
      DO 49 I=1,4
      DO 49 K=1,4
      FX2(I,K)=F1X2(I,K)+F2X2(I,K)+F3X2(I,K)+F4X2(I,K)
 49   CONTINUE
      DO 50 I=1,4
      DO 50 K=1,4
      SDHX(I,K)=SDJX(I,K)+FX2(I,K)
 50   CONTINUE
C     TYPE*,J,((SDHX(I,K),I=1,4),K=1,4)
      RETURN
      END
C-----------------------------------------------------------
C
C     THIS SUBROUTINE SOLVES THE "Q" DIFFERENTIAL EQUATIONS USING RUNGE
C     -E-KUTTA METHOD
      SUBROUTINE RUNG4(ALPHA2,ALPHA3,A)
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001),TN(2001)
      DIMENSION Q(4,1,101),QORG1(4,1),QR(4,1),QORG(4,1)
      DIMENSION CQ1(4,1),CQ2(4,1),CQ3(4,1),CQ4(4,1)
      DIMENSION CQ(4,1),FQ(4,1)
      DIMENSION EO(9),ZO(9)
      DIMENSION HT2(101),HT1(101),DFTP(1,4,101),R(1,4,101)
C-----------------------------------------------------------
      COMMON/A2/X/A3/Z
      COMMON/A7/V/A6/TN
      COMMON/A9/EGO/E1/EO,ZO,RG
      COMMON/A22/QR/A23/FQ/A24/Q/FF1/PT,N
      COMMON/A26/HT1/A18/HT2/B15/DFTP/A16/R
C-----------------------------------------------------------
      REAL MP
C     A=A/5.0
      DO 300 JI=1,100
```

```
         J1=102-JI
         J=J1
         I=1
         DO 53 K=1,4
         QORG1(K,I)=Q(K,I,J)
53       CONTINUE
C        M=1
         N=0
C        B=0.0
5555     CONTINUE
         DO 55 K=1,4
         QR(K,I)=Q(K,I,J)
         QORG(K,I)=Q(K,I,J)
55       CONTINUE
         L=((J-1)*20)-N+1
C        T=TN(L)-B*(TN(L)-TN(L-1))
         T=TN(L)
         DO 1 I=3,9
         Z(I,J)=60.0*EXP(-EO(I)/(RG*T)+ZO(I))
1        CONTINUE
         X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
         X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
         X3=(19.422+0.025*(T-413.0))/1000.0
         MP=(X(1,L)+X(2,L)+X(3,L)+X(4,L))/2.0
         VTO=X1*MP+X2*X(5,L)+X3*X(6,L)
         DO 1111 I=3,9
         IF((I.EQ.4).OR.(I.EQ.9))GOTO 1111
         Z(I,J)=Z(I,J)/VTO
1111     CONTINUE
C        B=B+0.2
         CALL EQTN4(ALPHA2,ALPHA3,J,L,T)
         DO 202 K=1,4
         CO1(K,I)=A*FQ(K,I)
         QR(K,I)=QORG(K,I)-0.5*CO1(K,I)
202      CONTINUE
         CALL EQTN4(ALPHA2,ALPHA3,J,L,T)
         DO 203 K=1,4
         CO2(K,I)=A*FQ(K,I)
         QR(K,I)=QORG(K,I)-0.5*CO2(K,I)
203      CONTINUE
         CALL EQTN4(ALPHA2,ALPHA3,J,L,T)
         DO 206 K=1,4
         CO3(K,I)=A*FQ(K,I)
         QR(K,I)=QORG(K,I)-CO3(K,I)
206      CONTINUE
         CALL EQTN4(ALPHA2,ALPHA3,J,L,T)
         DO 205 K=1,4
         CO4(K,I)=A*FQ(K,I)
         CO(K,I)=(CO1(K,I)+2.0*(CO2(K,I)+CO3(K,I))+CO4(K,I))/6.0
         Q(K,I,J)=QORG(K,I)-CO(K,I)
205      CONTINUE
C        IF(M.LE.5)GOTO 5555
         N=N+1
C        B=0.0
C        M=1
         IF(N.LE.19)GOTO 5555
         DO 56 K=1,4
         Q(K,I,J-1)=Q(K,I,J)
         Q(K,I,J)=QORG1(K,I)
56       CONTINUE
300      CONTINUE
```

```
C
779     A=A*5.0
        RETURN
        END
C
C       SUBROUTINE TO SOLVE "Q" DIFFERENTIAL EQUATIONS
        SUBROUTINE EQTN4(ALPHA2,ALPHA3,J,L,T)
        DIMENSION X(7,2001),V(4,2001)
        DIMENSION Z(9,2001)
        DIMENSION QR(4,1),EQ(4,1)
        DIMENSION DFT(1,4),HT1(101),HT2(101),DFTP(1,4,101)
        DIMENSION RTS(4,1),DFTN(1,4),RRNEW(4,1)
        DIMENSION XFQ(4,1),R(1,4,101),RP(1,4),TRR(4,1)
        DIMENSION FX(4,4),XF(4,4),EO(9),ZO(9)
C
        COMMON/A2/X/A3/Z/E1/EO,ZO,RG
        COMMON/A7/V
        COMMON/A9/EGO/A15/DFT/A26/HT1/B15/DFTP
        COMMON/A16/R
        COMMON/A18/HT2/A19/FX,XF
        COMMON/A22/QR/A23/FQ/FF1/PT,H
C
        CALL FXF(J,L,T)
        DO 202 I5=1,4
        DFT(1,I5)=DFTP(1,I5,J)
202     CONTINUE
        K=1
        DO 8 I=1,4
        RR(K,I)=R(K,I,J)
        TRR(I,K)=RR(K,I)
8       CONTINUE
        DO 9 I=1,4
        RTS(I,K)=TRR(I,K)*HT2(J)*HT1(J)
9       CONTINUE
C       MULTIPLICATION OF THE MATRICES "DFT" AND "QR"
        DO 10 I=1,4
        DFTN(K,I)=HT2(J)*DFT(K,I)
10      CONTINUE
        DFTQ=0.0
        DO 11 I=1,4
        DFTQ=DFTQ+DFTN(K,I)*QR(I,K)
11      CONTINUE
        DO 12 I=1,4
        RRNEW(I,K)=TRR(I,K)*DFTQ
12      CONTINUE
C       MULTIPLICATION OF THE MATRICES "FXF" AND "QR"
        I1=1
        DO 14 K1=1,4
        XFQ(K1,I1)=0.0
        DO 15 L1=1,4
        XFQ(K1,I1)=XFQ(K1,I1)+XF(K1,L1)*QR(L1,I1)
15      CONTINUE
14      CONTINUE
        DO 16 I=1,4
        FQ(I,K)=RTS(I,K)+RRNEW(I,K)-XFQ(I,K)
16      CONTINUE
        RETURN
        END
C
C       THIS SUBROUTINE INTEGRATES THE "DELLX" DIFFERENTIAL EQUATIONS
C       FORWARD W.R.T. "TIME"
        SUBROUTINE RUNG5(ALPHA2,ALPHA3,A)
```

```fortran
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001),TN(2001)
      DIMENSION DORG1(4,1),DELLX(4,1,101)
      DIMENSION DELXRG(4,1),DELLXR(4,1),TOWX(4,1),EO(9),ZO(9)
      DIMENSION CD1(4,1),CD2(4,1),CD3(4,1),CD4(4,1),CD(4,1)
      DIMENSION R(1,4,101),RR(1,4),O(4,1,101),QP(4,1)
      DIMENSION HT1(101),HT2(101),DFTP(1,4,101)
C
      COMMON/A2/X/A3/Z/E1/EO,ZO,RG
      COMMON/A7/V/A6/TN
      COMMON/A9/EGO
      COMMON/A25/DELLX/A27/DELLXR
      COMMON/A28/TOWX/FF1/PT,H
      COMMON/A16/R/A24/O/A26/HT1/A18/HT2/B15/DFTP
      REAL MP
C
      A=A/5.0
      DO 100 J1=1,100
      J=J1
      L=1
      DO 99 K=1,4
      DORG1(K,L)=DELLX(K,L,J)
99    CONTINUE
C     NNN=1
      N=0
C     B=0.0
5555  CONTINUE
      DO 65 K=1,4
      DELLXR(K,L)=DELLX(K,L,J)
      DELXRG(K,L)=DELLX(K,L,J)
65    CONTINUE
      M=((J-1)*20)+N+1
C     T=TN(M)+B*(TN(M+1)-TN(M))
      T=TN(M)
      DO 11111 I=3,9
      Z(I,J)=60.0*EXP(-EO(I)/(RG*T)+ZO(I))
11111 CONTINUE
      X1=191.5*(1.0+0.0014*(T-413.0))/1000.0
      X2=60.6*(1.0+0.0014*(T-413.0))/1000.0
      X3=(19.422+0.025*(T-413.0))/1000.0
      MP=(X(1,M)+X(2,M)+X(3,M)+X(4,M))/2.0
      VTO=X1*MP+X2*X(5,M)+X3*X(6,M)
      DO 22222 I=3,9
      IF((I.EQ.4).OR.(I.EQ.9))GOTO 22222
      Z(I,J)=Z(I,J)/VTO
22222 CONTINUE
C     B=B+0.2
      CALL EQTN5(J,M,T)
      DO 902 K=1,4
      CD1(K,L)=A*TOWX(K,L)
      DELLXR(K,L)=DELXRG(K,L)+0.5*CD1(K,L)
902   CONTINUE
      CALL EQTN5(J,M,T)
      DO 903 K=1,4
      CD2(K,L)=A*TOWX(K,L)
      DELLXR(K,L)=DELXRG(K,L)+0.5*CD2(K,L)
903   CONTINUE
      CALL EQTN5(J,M,T)
      DO 906 K=1,4
      CD3(K,L)=A*TOWX(K,L)
      DELLXR(K,L)=DELXRG(K,L)+CD3(K,L)
```

```
906   CONTINUE
      CALL EQTN5(J,M,T)
      DO 905 K=1,4
      CD4(K,L)=A*TOWX(K,L)
      CD(K,L)=(CD1(K,L)+2.0*(CD2(K,L)+CD3(K,L))+CD4(K,L))/6.0
      DELLX(K,L,J)=DELXRG(K,L)+CD(K,L)
905   CONTINUE
C
      IF(NNN.LE.5)GOTO 5555
      N=N+1
      NNN=1
      B=0.0
      IF(N.LE.19)GOTO 5555
      DO 66 K=1,4
      DELLX(K,L,J+1)=DELLX(K,L,J)
      DELLX(K,L,J)=DORG1(K,L)
66    CONTINUE
100   CONTINUE
C
      A=A*5.0
      RETURN
      END
C-------------------------------------------------------------------
C     SUBROUTINE TO SOLVE "DELLX" DIFFERENTIAL EQUATIONS
      SUBROUTINE EQTN5(J,M,T)
      DIMENSION X(7,2001),V(4,2001)
      DIMENSION Z(9,2001),E0(9),Z0(9)
      DIMENSION DELLXR(4,1),TOWX(4,1)
      DIMENSION R(1,4,101),RR(4,4),Q(4,1,101),QR(4,1)
      DIMENSION HFX(4,4),HT2(101),FX(4,4),XFDX(4,1)
      DIMENSION XF(4,4),DFT(1,4),HT1(101),FDT(4,1),DFTP(1,4,101)
C-------------------------------------------------------------------
      COMMON/A2/X/A3/Z/E1/E0,Z0,RG
      COMMON/A7/V/A15/DFT/A26/HT1
      COMMON/A9/EGO/A16/R/A18/HT2/A19/FX,XF
      COMMON/A27/DELLXR/A24/Q
      COMMON/A28/TOWX/FF1/PT,H/B15/DFTP
C-------------------------------------------------------------------
      CALL FXF(J,M,T)
      DO 202 I5=1,4
      DFT(1,I5)=DFTP(1,I5,J)
202   CONTINUE
C     MULTIPLICATION  OF THE MATRIX "FDT" AND "HT2"
      DO 89 I=1,4
      FDT(I,1)=DFT(1,I)
89    CONTINUE
      DO 10 I=1,4
      HFX(I,1)=HT2(J)*FDT(I,1)
10    CONTINUE
      L=1
      DO 19 I=1,4
      RR(L,I)=R(L,I,J)
      QR(I,L)=Q(I,L,J)
19    CONTINUE
C     MULTIPLICATION OF THE MATRICES "XF" AND "DELLX"
      DO 991 J3=1,4
      XFDX(J3,L)=0.0
      DO 992 K3=1,4
      XFDX(J3,L)=XFDX(J3,L)+XF(J3,K3)*DELLXR(K3,L)
992   CONTINUE
991   CONTINUE
C     MULTIPLICATION OF THE MATRICES "RR" AND "DELLX"
      RRDX=0.0
```

```fortran
      DO 994 K=1,4
      RRDX=RRDX+RR(L,K)*DELLXR(K,L)
994   CONTINUE
C     MULTIPLICATION OF THE MATRICES "DFT" AND "QR"
      DFTQ=0.0
      DO 993 K5=1,4
      DFTQ=DFTQ+DFT(L,K5)*QR(K5,L)
993   CONTINUE
      SUM=0.0
      SUM=SUM+HT1(J)+RRDX+DFTQ
      DO 995 I=1,4
      HFX(I,1)=HFX(I,1)*SUM
995   CONTINUE
      DO 15 I=1,4
      TOWX(I,L)=XFDX(I,L)-HFX(I,L)
15    CONTINUE
      RETURN
      END
C
C*****************************************************************
C     THIS SUBROUTINE INTEGRATES THE INTEGRAL TERM IN THE OBJECTIVE
C     FUNCTION USING SIMPSONS RULE
      SUBROUTINE SIMPSN
      DIMENSION X(7,2001),SX(6,2001)
      COMMON/A1/SSX(6)/A2/X
      COMMON/A9/EGO
      DO 1008 K=1,4
      SSX(K)=0.0
1008  CONTINUE
      DO 666 I=2,4,2
      DO 99 J=3,1999,2
      SX(I,J)=SSX(I)+2.0*(X(I,J)*X(I,J))
      SSX(I)=SX(I,J)
99    CONTINUE
      DO 108 J=2,2000,2
      SX(I,J)=SSX(I)+4.0*(X(I,J)*X(I,J))
      SSX(I)=SX(I,J)
108   CONTINUE
      SSX(I)=(1/6003.0)*(SSX(I)+(X(I,1)**2)+(X(I,2001)**2))
666   CONTINUE
      DO 5000 J=3,1999,2
      SX(1,J)=SSX(1)+2.0*((EGO/X(1,J))-15.0)*((EGO/X(1,J))-15.0)
      SSX(1)=SX(1,J)
5000  CONTINUE
      DO 5001 J=2,2000,2
      SX(1,J)=SSX(1)+4.0*((EGO/X(1,J))-15.0)*((EGO/X(1,J))-15.0)
      SSX(1)=SX(1,J)
5001  CONTINUE
      SSX(1)=(1.0/6003.0)*(SSX(1)+((EGO/X(1,1))-15.0)**2)+
     1(((EGO/X(1,2001))-15.0)**2))
      RETURN
      END
C*****************************************************************
```